Aston University

If you have discovered material in AURA which is unlawful e.g. breaches copyright, (either yours or that of a third party) or any other law, including but not limited to those relating to patent, trademark, confidentiality, data protection, obscenity, defamation, libel, then please read our Takedown Policy and contact the service immediately

ADVANCED HANDOVER PROCEDURE FOR CELLULAR COMMUNICATION SYSTEMS

AHMED HADI ALI

Doctor of Philosophy

ASTON UNIVERSITY

OCTOBER 1998

This copy of the thesis has been supplied on condition that anyone who consults it is understood to recognise that its copyright rests with its author and that no quotation from the thesis and no information derived from it may be published without proper acknowledgment.

Aston University

ADVANCED HANDOVER PROCEDURE FOR CELLULAR COMMUNICATION SYSTEMS

AHMED HADI ALI

Submitted for the Degree of Doctor of Philosophy

1998

Summary

Third Generation cellular communication systems are expected to support mixed cell architecture in which picocells, microcells and macrocells are used to achieve full coverage and increase the spectral capacity. Supporting higher numbers of mobile terminals and the use of smaller cells will result in an increase in the number of handovers, and consequently an increase in the time delays required to perform these handovers. Higher time delays will generate call interruptions and forced terminations, particularly for time sensitive applications like real-time multimedia and data services.

Currently in the Global System for Mobile communications (GSM), the handover procedure is initiated and performed by the fixed part of the Public Land Mobile Network (PLMN). The mobile terminal is only capable of detecting candidate base stations suitable for the handover; it is the role of the network to interrogate a candidate base station for a free channel. Handover signalling is exchanged via the fixed network and the time delay required to perform the handover is greatly affected by the levels of teletraffic handled by the network.

In this thesis, a new handover strategy is developed to reduce the total time delay for handovers in a microcellular system. The handover signalling is diverted from the fixed network to the air interface to prevent extra delays due to teletraffic congestion, and to allow the mobile terminal to exchange signalling directly with the candidate base station. The new strategy utilises Packet Reservation Multiple Access (PRMA) technique as a mechanism to transfer the control of the handover procedure from the fixed network to the mobile terminal. Simulation results are presented to show a dramatic reduction in the handover delay as compared to those obtained using fixed channel allocation and dynamic channel allocation schemes.

Key words: TDMA, PRMA, GSM, Handover, Multiple access techniques.

ACKNOWLEDGEMENTS

I would like to thank my supervisor Dr. R. L. Brewster for all the encouragement and constructive criticism he has given throughout the research, especially after the start of his retirement. Many thanks to Professor M. J. Cardwell for his supervision and continuous attention to my research during the final year.

Special thanks to Dr. Jaspal Sagoo for his time and effort in revising and proof reading the thesis throughout the writing process. Also, I would like to thank Dr. David Bozward for his efforts in providing technical advice and documentation. Many thanks to Mrs. Edna Archer for all the relevant administrative work she carried out during my research.

I am so grateful to my father, my mother and my sister for their generosity, support and faith which gave me the strength to complete this work.

To my most precious,
my Mother and my Father

CONTENTS

Chapter Four: Simulation and Modelling

Chapter Five: Advanced Handover Signalling Strategy

Chapter Six: Discussion, Conclusion and Future Work Recommendations

Abbreviations

AMPS	Advanced Mobile Phone System
ATDMA	Advanced Time Division Multiple Access
BSC	Base Station Controller
BSIC	Base Station Identity Code
BSS	Base Station Sub-system
BSSAP	Base Station Sub-system Application Part
CDMA	Code Division Multiple Access
CIR	Carrier to Interference Ratio
CT-2	Cordless Telephone (second generation)
DCA	Dynamic Channel Allocation
DCCH	Dedicated Control Channel
DCS	Digital Communication Service
DECT	Digital European Cordless Telecommunication system
ERMES	European Radio Message Service
FACCH	Fast Associated Control Channel
FCA	Fixed Channel Allocation
FDMA	Frequency Division Multiple Access
GEO	Geostationary Earth Orbiting
GSM	Global System for Mobile communications
IMT-2000	International Mobile Telecommunications-2000
ITU	International Telecommunications Union
LEO	Low Earth Orbiting
MSC	Mobile Switching Center
NMT-450	Nordic Mobile Telephone System-450
OSI	Open System Interface
PACS	Personal Access Communication System
PCS	Personal Communication Service
PLMN	Public Land Mobile Network
PRMA	Packet Reservation Multiple Access
QoS	Quality of Service
RSSI	Received Signal Strength Intensity
TACS	Total Access Communication System
TCH	Traffic Channel
TDMA	Time Division Multiple Access
VLR	Visitor Location Register

List of Tables

List of Figures

CHAPTER ONE

INTRODUCTION

1.1 Introduction

The research presented in this thesis introduces a new method to reduce the handover delay and improve the overall handover performance in Third Generation mobile communication systems. This is achieved by the introduction of a handover signalling strategy that bypasses the fixed part of the Public Land Mobile Network (PLMN) and transfers the handover control to the mobile terminal. The new method utilises the Packet Reservation Multiple Access ++ (PRMA++) as a medium access technique to implement the new strategy.

First Generation cellular communication systems have been in use since the launch of Advanced Mobile Phone Systems (AMPS) in the USA in 1978 [1]. This was followed by a joint development in the four Nordic countries (Denmark, Finland, Norway, and Sweden) to introduce the Nordic Mobile Telephone system (NMT450) in 1981 operating in the 450 MHz band. In 1985 the Total Access Communication System (TACS) was adopted in the UK, which was a modified version of the AMPS standard [2]. These analogue systems provided services with the opportunity to roam freely within the coverage area and simultaneously communicate with any phone , fax or data modem subscriber anywhere. However, the advance in digital technology led to the introduction of the Second Generation mobile communication systems. Specifically the GSM (Global System for Mobile communications) was introduced in 1989 as a Second Generation pan-European digital mobile radio communication system operating in the region of the 900 MHz frequency band. Using digital technology the GSM standard was capable of offering a number of new features like:

- integration of voice and data,
- efficient utilisation of the limited radio space as compared with the First Generation systems,
- providing better speech quality and higher data rate,
- allowing international roaming and the provision of service in any GSM network regardless of the country providing the network,

Currently, GSM is considered to be the most established digital cellular communication system, which is used in Europe and other parts of the world. Also widely adopted is the Digital Communication Services -1800 (DCS-1800), which is based on the GSM standard but operates in the 1800 MHz frequency band and has higher capacity allocation than GSM. The Code Division Multiple Access (CDMA) standard, which uses a spread spectrum technique, had proven to be one of the world's most successful Second Generation cellular systems and it is widely used in North America [3].

The cellular communication services are likely to be based on combinations of technical standards from the Second Generation mobile communication systems (e.g. DECT, GSM, DCS-1800, CDMA, ERMES, MOBITEX, and HiperLAN), providing a service in many different parts of the radio frequency spectrum (cordless telephone, cellular, paging, mobile data, and wireless LAN). Ultimately, cellular communication services should be carried via a single integrated technology concept that is accessible by all the different standards [4,5,6,7]. Also the rapid development in telecommunication networks, coupled with an expanding base of mobile communication subscribers, is producing a continuous need for new types of wireless services. Hence, there is a genuine need for a unified standard for future wireless communication systems, and currently it is the driving force behind the development of Third Generation mobile communication systems.

Future wireless services are expected to support not only traditional mobile voice communications, but also a variety of voice and data services providing a wide range of applications. Specifically, it is expected that services such as multimedia capabilities, Internet access, imaging and video conferencing will be supported by any Third

Generation wireless system. The International Telecommunication Union (ITU) have specified the need for Third Generation mobile systems to support advanced high-bit-rate services. A variable-rate access with data rates approaching 2 Mb/s has been proposed for the International Mobile Telecommunications-2000 (IMT-2000), formerly known as Future Public Land Mobile Telecommunications Systems (FPLMTS) [8]. Third Generation cellular communication systems are also expected to provide global telecommunications services. This requires contiguous coverage throughout the globe and the capability of accessing any cellular network at any geographical location. Satellite-based mobile systems can provide ubiquitous global services to mobile terminals, particularly in areas where constructing terrestrial networks is not economical or impossible due to natural boundaries. While in densely populated areas, terrestrial-based systems can provide high capacity through the use of microcells and picocells, which give high spectrum reuse and allow the penetration of cellular coverage into indoor and enclosed areas. Such areas would otherwise be shadowed from the top-of-the-roof antennae transmission due to the urban structures, in both streets and building environments.

1.2 Handover in The Microcellular Environment

The most efficient method to increase the network capacity is the introduction of microcells with radii of about 0.5 km down to 100m and below. Microcells extend the cellular coverage area from macrocells to shadowed street corners, enclosed areas inside buildings, road and railway tunnels, and even underground buildings and train stations. In such areas a microcellular base station can be the only interface available for the mobile terminal to access the cellular network (i.e. there are no overlaying macrocells). However, as more users are supported by introducing smaller cell sizes there will be an increase in the number of handovers taking place, especially with terminals moving at high speed through shrinking cells. The success of a handover process relies on a number of factors; the most important factors include: (1) the availability of a free channel in the target cell; (2) the number of alternative base stations, if no free channels are available in the first cell; (3) the time limit allowed to carry out a handover procedure before the call is dropped. Any increase in the number of border crossings during the call will significantly increase the probability of call dropping due to failure in acquiring free spectrum allocations. Also, the increase in the number of mobile terminals handing over simultaneously will impose great pressure on the signalling infrastructure of the network and increase the probability of failure due to insufficient network resources.

Third Generation mobile communication systems will require efficient and reliable handover procedures to maintain the continuity and quality of the call (or the communication) while the mobile terminal is roaming within the same network or between two different networks. The research carried out in this thesis is based on the premise that with the existing network infrastructure, current handover procedures are unlikely to be capable of handling new services with higher bit rates and strict delay limits. Hence, faster handover execution and more efficient utilisation of network resources is a prime requirement in any future network, particularly where microcells are deployed.

1.3 Research Objectives

Third Generation systems are essentially evolving from the Second Generation mobile communications. Such a service evolution is expected to be relatively seamless, attractive, and natural for the current user markets [4]. Therefore the introduction of future systems must carry a certain degree of compatibility with the older systems, and achieving this will entail the partial or full re-use of the Second Generation networks infrastructure. The GSM standard is the most feature-rich cellular digital system for the Second Generation mobile communications, with most operational networks supporting basic voice and data services. Therefore, GSM is expected to represent an evolution platform for many of the Third Generation services, and consequently it was decided that the GSM standard and network infrastructure will be considered solely for the investigations and simulations carried out in this research work. The research objectives can be summarised as follows:

(i) The GSM standard stipulates that a handover procedure is performed by exchanging the handover signalling between the old base station and the new base station via the fixed part of the Public Land Mobile Network (PLMN). However, as more users are supported and smaller cells are deployed, the success of the handover process can be greatly affected by the limitations of switching and signalling infra-structure in the PLMN. One of the objectives of this research is to assess the handover performance of a GSM-based microcellular system. This assessment will focus on examining the total time delay required to perform the handover process utilising a number of dynamic channel allocation (DCA) schemes, and subsequently comparing their performance to the fixed channel allocation (FCA) schemes already implemented in GSM.

(ii) An improvement of the handover performance within the GSM will be considered by developing a strategy that introduces major changes to the functional roles assigned to the network and the mobile terminal. This new handover strategy will involve the diversion of the handover signalling from the fixed part of the PLMN to the air interface via the mobile terminal.

(iii) The Packet Reservation Multiple Access ++ (PRMA++) scheme has been previously developed as an access method to offer packet based resource assignment in the Advanced TDMA (ATDMA) project [9,10]. In order to reduce the handover delays and increase the probability of successful handovers in a microcellular environment, PRMA++ will be investigated for the possibilities of extending its operation to transmit user information and in the same time maintain network-independent handover.

1.4 Thesis Overview

Chapter Two of this thesis introduces the fundamentals of the handover process. In particular it focuses on the process of initiating a handover in a network, and the metrics used to evaluate the handover performance in certain systems. Moreover, a review of the main handover strategies suggested and implemented, including network controlled, mobile assisted, and mobile controlled handover, is presented. The use of handovers in the context of microcellular environments is also considered, as well as the effects of multiple call handovers on the overall performance of the system. The chapter concludes by providing a discussion of integrated systems. These systems utilise mobile satellite systems to act as a complementary component to the terrestrial cellular network as part of the global cellular communication systems.

Chapter Three investigates the various types of radio interface multiple access techniques. Specifically, FDMA, TDMA, CDMA, and PRMA. However, this survey concentrates on the PRMA technique and reveals the inherent characteristics that make it suitable for the implementation of the advanced handover strategy over the radio interface. This chapter also presents a review of FCA and DCA techniques, and their effects on improving the spectrum efficiency and increasing the probability of successful handovers. The chapter concludes by considering the advantages and costs associated with the implementation of each scheme for satisfying efficient handover requirements.

Chapter Four outlines the simulation methods and software used in this research to simulate the cellular network and the medium access protocols. It starts by specifying the OSI model layer that is of specific interest to this research, it then moves on to discuss the COMNET III software which was used to determine delays for individual speech and handover signalling packets over the fixed part of the PLMN. The chapter explains the advantages and limitations of using COMNET III. The last section discusses the use of C++ code programming to simulate the new handover strategy and the different multiple access techniques in the microcellular network. A detailed description of the network models and parameters is presented in chapter Five.

Chapter Five introduces an advanced handover signalling strategy to reduce the total handover delay by diverting the handover signalling from the fixed part to the air-interface of the PLMN. A detailed description of the handover procedure in the GSM standard is presented, followed by a description of the improved handover signalling and how the control of the handover procedure is transferred to the mobile terminal to achieve a network-independent handover. Using COMNET III software, a GSM system with microcellular base stations is simulated to examine the delays encountered by single handover and speech packets. In the following section a microcellular network with 14 microcells is simulated for TDMA operation with FCA and DCA schemes. The signalling mechanism in the advanced handover signalling strategy is discussed and then followed by a comparison between the performance of PRMA++ and that of the advanced strategy using the same 14 microcellular network. Results obtained show a considerable reduction in the total handover time delay and a reliable overall performance as compared to the PRMA++ technique and the GSM standard.

Chapter Six concludes with a review and discussion of the work presented in this thesis. The advanced signalling strategy is compared against earlier handover policies and current cellular systems are evaluated for their suitability to implement the new strategy. Recommendations for future work highlights the potential of improvement in this strategy, and the scope of changes required to extend the concept of network-independent handovers beyond the borders of a single network.

CHAPTER TWO

HANDOVER PRINCIPLES AND TECHNIQUES

2.1 Introduction

Handover is the mechanism that transfers an ongoing call from one cell to another as a user moves through the coverage area of a cellular system. In analogue systems [3], the base station monitors the quality of the link between a mobile terminal and itself. When the base station realises that the quality of the link has degraded and the distance to the mobile terminal has become too large, it requests the adjacent cells to report the power level they see for the mobile back to the network. It is reasonable that the strongest reported power level for the mobile terminal comes from the closest base station to the mobile terminal. The network then decides which frequency channel the base station should use in the new cell and which corresponding frequency the mobile terminal should tune to. Eventually, the mobile terminal is commanded to perform a channel change.

The situation in GSM is different [11]; the mobile terminal must continuously monitor the neighbouring cells, perceived power levels. To do this, the base station gives the mobile terminal a list of base stations (channels) on which to perform power measurements. The measurement results are put into a measurement report, which is periodically sent back to the base station.

The base station itself may also be performing measurements on the power and quality of the link to the mobile terminal. If these measurements indicate the necessity of a handover, this can be performed without delay, as the appropriate new base station is already known. The measurements are sent continually to the base station and they reflect the mobile's point of view. It is up to the operator to act upon different qualities or power levels, and the handover constraints or thresholds can be adjusted in accordance with changing environment and operating conditions. [3]

In future systems, smaller cells (named pico or microcells) will be deployed to meet the demands for increased capacity [12,13]. This will result in an increase in the number of cell boundary crossings. Each handover requires network resources to re-route the call to the new base station. Minimising the expected number of handovers minimises the switching load. During the handover there is a brief service interruption. As the frequency of these interruptions increases, the quality of service (QoS) is reduced. Another concern is delay. If handover does not occur quickly, the QoS may degenerate below an acceptable level. The chances of dropping a call due to factors such as the availability of channels, increase with the number of handover attempts. All of these issues place additional challenges on the cellular system.

In this chapter, a general review of the handover process in mobile cellular systems is presented. Also presented is a description of the various techniques and research work aimed at improving the handover procedure in the different cellular environments. Handover requirements in multi-layered and integrated cellular systems are discussed at the end of the chapter.

2.2 Handover Initiation

Beyond a certain point, where the received signal is too weak or is dangerously close to becoming too weak, a handover is required. Measurements of received signal strength must be averaged over time to remove the rapid fluctuations due to multipath propagation. Detailed studies [14,15] have concluded that handover initiation criteria are based on essentially four variables:

(i) The length of the averaging window.

(ii) The shape of the averaging window (that is, how much we should trust older measurements).

(iii) A threshold level which may provide a trigger to commence a handover.

(iv) A hysteresis margin.

A key issue in the design of a handover algorithm is preventing a user from experiencing bouncing back and forth between two base stations. Two approaches to achieve this are the hysteresis margin and dwell-timer approaches. The former only allows a handover to a base station which is stronger than the current one by at least a fixed or time varying hysteresis margin. The latter restricts the minimum time between handovers. Figure (2.1) shows a mobile terminal moving from one base station (base 1) to another (base 2). The averaged signal strength of base 1 decreases as the mobile moves away from it. Similarly the averaged signal strength of base 2 increases as the mobile approaches it. Using this figure, the following approaches in handover initiation can be explained:

- *Relative signal strength*: chooses the strongest received base station at all times. In Figure (2.1) the handover will occur at position A. The problem with this method is that it tends to stimulate too many unnecessary handovers when the current base station signal is still adequate.

- *Relative signal strength with threshold*: allows the user to handover only if the current signal is sufficiently weak (less than a threshold) and the other base station's signal is the stronger of the two. The effect of the threshold depends on its value compared to the signal strengths of the two base stations at the point at which they are equal. If the threshold is higher than this value, say T_1 in Figure (2.1), this schemes performs exactly like the relative signal strength scheme, so the handover occurs at position A. If the threshold is lower than this value, say T_2 in Figure (2.1), the mobile will delay handover until the current signal level crosses the threshold at position B. In the case of T_3, the delay may be so long that the mobile terminal drifts far into the new cell. This reduces the quality of the communication link and may result in a dropped call. In addition, this causes additional interference to co-channel users. Thus, this scheme may create overlapping cell coverage areas. A threshold is not used alone in practice because its effectiveness depends on prior knowledge of the crossover signal strength between the current and candidate base stations.

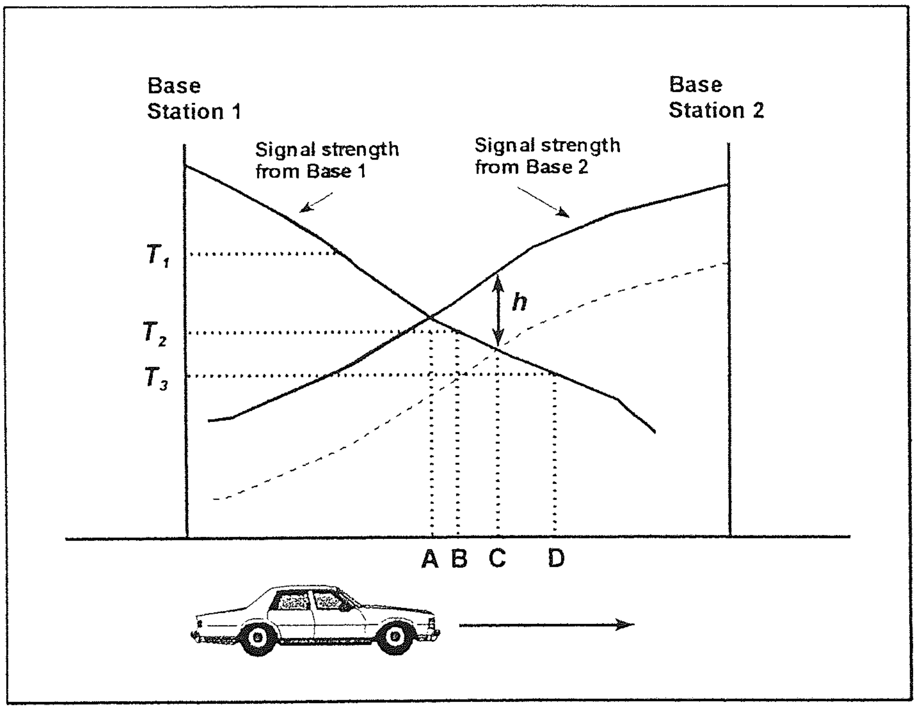


Figure (2.1): Trends in approaches to handover initiation.

- *Relative signal strength with hysteresis*: allows a user to handover only if the new base station is sufficiently stronger (by a hysteresis margin, h, in Figure (2.1)) than the current one. In this case the handover will occur at point C. This technique prevents the so-called Ping-Pong effect, the repeated handover between two base stations caused by the rapid fluctuations in the received signal strengths from both base stations. The first handover, however, may be unnecessary if the serving base station is sufficiently strong.
- *Relative signal strength with hysteresis and threshold*: hands a user over to a new base station only if the current signal level drops below a threshold and the target base station is stronger than the current base station by a given hysteresis margin. In Figure (2.1) the handover will occur at point C if the threshold is either T_1 or T_2, and will occur at point D if the threshold is T_3.

- *Prediction techniques*: base the handover decision on the expected value of the received signal strength. A technique which was proposed in [16] has been shown to perform better than both the relative signal strength and relative signal strength with hysteresis in terms of a reduction in the number of unnecessary handovers.

2.3 Handover Performance Metrics

To evaluate the handover performance of a cellular system, a number of performance metrics are used to assess the handover algorithms [15,17], these metrics are:

- *Call blocking probability*: The probability that a new call is blocked by the system.
- *Handover blocking probability*: The probability that a handover attempt is blocked.
- *Handover probability*: The probability, while communicating with a particular base station, that an ongoing call requires a handover before the call terminates. This metric translates into the average number of handovers per call.
- *Call dropping probability*: The probability that a call terminates due to a handover failure. This metric can be derived directly from the handover blocking probability and the handover probability.
- *Probability of an unnecessary handover*: The probability that a handover is stimulated by a particular handover algorithm when the existing radio link is still adequate.
- *Duration of interruption*: The length of time during a handover for which the mobile terminal is communicating with neither base station. This metric is heavily dependent on the particular network topology and the scope of the handover [18].
- *Delay*: The time interval commencing when the mobile terminal starts a handover procedure, and ending when the mobile is allocated a new link with the new base station. This metric is crucial because excessive delay will affect the

quality of the communication link, and increase the call dropping probability and the duration of interruption.

2.4 Handover Types

There are two types of handovers according to the mechanism used to connect and disconnect new and old links, respectively. The first type is Soft handover, during which the mobile terminal is in communication with the old as well as the new base station. The connection is maintained with both until the handover procedure is complete. The second type is the Hard handover, which occurs when the link with the old base station is disconnected before a new connection is activated with the new base station. The first type is only feasible when there is enough spectral capacity in the system to support simultaneous communications on two channels. However, hard handover is more likely to result in a service interruption if there is an increase in the time required to allocate the new channel.

Soft handover is utilised in the Digital Enhanced Cordless Telecommunication system (DECT) [19]. As the mobile terminal is moving, the status of all available channels is updated and when a handover is needed the terminal selects the most suitable voice channel and makes a second connection while it is still in conversation. The handset simply instructs the base station to connect the new voice path in parallel with the existing one and subsequently to close down the old voice channel. In GSM [20], however, the mobile terminal is in communication with the old base station while the new channel is being determined. Once the mobile is assigned a channel, the old base station is instructed to release the old connection and the call is re-routed to the new base station. The new channel is determined, but not connected until the link with the first base station has been cleared.

2.5 Handover Control Strategies

Current cellular systems employ three major strategies to control the handover process. These differ in the way tasks and decision making responsibilities are distributed between the network and the mobile terminal. These strategies are as follows[21,22]:

2.5.1 Network Controlled Handover (NCHO): Employed by AMPS and Second Generation Cordless Telephone (CT-2 Plus) [23,24]. In this method the mobile terminal is completely passive. The base station supervises the quality of the current connection, i.e. the Received Signal Strength Intensity (RSSI) from the mobile terminal. RSSI measurements of alternative connections are also made by surrounding base stations. The decisions (when and where) are made at the Mobile Switching Centre (MSC). Only inter-cell handovers are possible. However, to reduce the signalling load in the network, the neighbouring base stations cannot send measurement reports continuously. Hence, updated measurements of neighbouring channels are not made very often and consequently the comparisons made will not take place as soon as the actual RSSI is below a certain threshold.

2.5.2 Mobile Assisted Handover (MAHO): This type is used in GSM where the handover process is more decentralised than NCHO. Both the mobile terminal and the base station supervise the received signal strength and the channel quality, i.e. RSSI and Bit Error Rate (BER). Measurements of alternative base stations transmission are made by the mobile terminal. The decisions, however, are still made in the fixed network (Base Station Sub-system (BSS) and MSC). Both inter-cell and intra-cell handovers are possible.

2.5.3 Mobile Controlled Handover (MCHO): This strategy is employed by the DECT air interface protocol [19]. In this method the mobile terminal continuously monitors the signal strength and channel quality from the current base station and several handover candidate base stations. When handover is required, the mobile terminal checks the best candidate base station for an available channel and launches a handover request. Both inter-cell and intra-cell handovers are possible.

Considering the increasing numbers of mobile subscribers and the continuous development in the types of services provided in cellular systems, a decentralised handover strategy is highly desirable to relieve the fixed network from the excessive signalling loads associated with high teletraffic levels.

2.6 Handover Management Schemes

In any handover strategy (NCHO, MAHO, or MCHO), the success of a handover process is dependent on two major factors: the first is the management of the available frequency allocations in every cell in the system, and the second is how fast a handover request is allocated the required resources. Efficient signalling also represents an important issue as the time required to carry out the handover will be affected by the level of traffic congestion in the network. This aspect is more pronounced in microcells where the cell size is reduced to increase the system spectral capacity [25,26]. Reducing the cell size will result in an increase in the number of handovers taking place. It will also reduce the time available for the mobile terminal to perform the handover, particularly in terminals moving at speed through the microcellular area. Therefore fast and efficient handover procedures are required. The following three sections review the different schemes suggested or implemented to perform handover in the cellular systems. The schemes considered in these sections deal with single call handover, multiple call handover, and handover between different layers of cellular structure within the same network.

2.6.1 Queuing and Priority

Lin *et al* [22] and Steele *et al* [27] investigated two basic schemes, the first is the Non-Prioritised Scheme (NPS), where the base station handles a handover request with the same priority as an originating call (i.e. the handover call is blocked immediately if no channel is available). This is the scheme employed by typical radio technologies for personal communication service (PCS) in the frequency allocation at 2 GHz. The second is the Reserved Channel allocation Scheme (RCS). In this scheme the channels are divided into two groups: normal channels that serve both new call attempts and handover calls, and reserved channels that serve handover calls only. For Reserved Channel Schemes, increasing the number of reserved channels will decrease the forced termination probability of an ongoing call. However, since the reserved channels cannot be assigned to new calls, increasing them will significantly increase the blocking probability of new call attempts [28].

Lin *et al* [22] also suggested a number of queuing and priority schemes based on the fact that in a cellular network, adjacent coverage areas would overlap, resulting in a considerable handover area where a call can be handled by the old and the new base stations. For a mobile-controlled handover, if a mobile terminal cannot find an available channel at the new base station, the handover request would be queued waiting for a channel. In this scheme, any released channel is allocated to a queuing handover request, which has priority over new call initiation. If there is no idle channel after the mobile terminal moves out of the handover area, then the call is forced to terminate. In the case of Personal Access Communication System (PACS) [29], a physical channel exists for the mobile terminal to signal to a blocked base station the handover attempt. In DECT, however, if a base station is blocked then no channel exists for the mobile terminal to make such a request. For network-controlled handover systems, e.g. CT-2 Plus, the old base station can always make such a request to the new base station. Service priority can be on a First-In-First-Out (FIFO) basis, where the next handover call to be serviced is the earliest one to arrive in the queue. In the Measured-Based Priority Scheme (MBPS) [30], the next handover call is selected based on the power level that the mobile terminal receives from the new base station. In this case the handover area can be viewed as regions marked by the different ranges of power ratio, and the network is expected to monitor the power levels of the queued handover calls dynamically.

It is crucial to notice that the performance of the queuing schemes discussed depends on the time that the mobile terminal stays in the handover area, also the time that the mobile terminal and the network will maintain the call connection while the radio link is down or unavailable. The queuing priority schemes add extra implementation complexity (compared to NPS) to manage the waiting queues. Considerable modifications to the base station and mobile terminal hardware/software would be required.

A scheme proposed by Fujii [31] which was based on the concept of overlapping cells is Selective Handover. In this scheme, if the traffic of a cell increases temporarily so that the resource utilisation rate exceeds a threshold, the scheme hands over some calls to the appropriate adjacent cells. Whenever a call reaches the overlapping area it can be served by the base station of either of the overlapping cells. The wider the cell overlapping, the more the traffic performance is expected to improve. Everitt [32] had also suggested a similar concept to implement a Directed Retry (DR), which takes advantage of the fact that some percentage of the mobile terminals may be able to obtain sufficient signal quality from two or more cells. If a mobile finds no free channels at its first-attempt cell, it can then try for a free channel in any other cell that can provide sufficient signal quality. The motivation of such schemes is to attempt to redistribute calls in heavily loaded cells to lighter loaded cells. These schemes improve traffic performance and can be considered superior to dynamic channel allocation (DCA) because it utilises the conventional intercell handover function and no new functions are necessary [31]. However, the performance improvement achieved by these schemes depends greatly on the algorithm used for selecting a mobile terminal for handover from a heavily loaded cell. Terminals chosen could be one of the following: mobile terminals with minimum reception level, i.e. the furthest away from the base station, or mobile terminals in the original cell that are capable of receiving an acceptable signal level from the adjacent cells which have one or more idle channels. In both cases the algorithms do not take into account co-channel interference.

A Sub-rating Scheme has also been suggested [33]. If the new base station has no idle channels, a new channel for a handover access attempt is created by sub-rating an existing call. Sub-rating means an occupied full rate channel is temporarily divided into two channels at half the original rate: one to serve the existing call and the other to serve the handover request. The PACS protocol allows for mobile terminals to indicate to the network the need to make a priority call [34]. This capacity is intended for emergency calls, maintenance calls and other priority calls such as handover requests when no traffic channel is available. Such a scheme has a number of limitations regarding the type of the traffic that can be carried by the sub-rated channel, and whether it tolerates delays or degradation in quality. For example voice quality for a sub-rated channel will not be as good as for a full rate channel, also coding size and time delay in real-time data transmission may require more dedicated sources than those already available.

Chia [35] introduced another strategy to enhance the robustness of the handover algorithm by implementing the concept of "forward" and "backward" handover. With a backward handover, handover signalling is performed via the serving base station. This is widely adopted in conventional cellular radio systems, but in microcells where the uplink carrier-to-interference ratio (CIR) could drop significantly when a mobile terminal turns a street corner [21] (see Figure.(2.2)), a backward handover process could fail. In order to circumvent this problem, a forward handover procedure could be introduced. With this procedure the handover signalling is performed via the new base station. Results showed improved performance for low speed mobile terminals passing through microcells-only coverage area. Handover success probability deteriorated with speed and high levels of traffic, considerable improvement obtained only in the case of using an "Umbrella" macrocell to assist in handovers. Such a strategy aims at recovering calls that fail a backward handover, and therefore allowing call re-establishment. However, higher time delays are very likely to occur, because the decision on the signalling destination (old or new station) is decided while the mobile terminal is suffering considerable degradation in the link quality. Also, a sudden improvement in the old base station link quality might give rise to a Ping-Pong effect [36] whereby the handover signalling control is bounced within the network. The success of this strategy is based on the assumption that the next base station, where the mobile terminal is heading, is the 1st candidate on the candidate cell list. Therefore, even though the uplink is deteriorating rapidly, the old base station is capable of starting the handover procedure with the target base station. The probability of success for such a strategy will be greatly reduced by the lack of free channels in the target base station, or by the mobile terminal heading towards a base station different from what the old base station predicts.

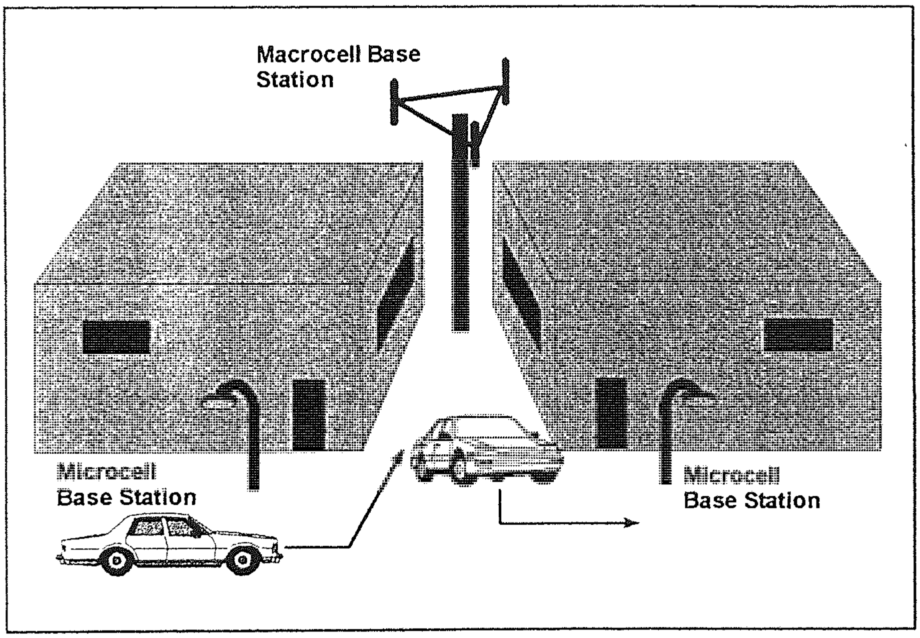


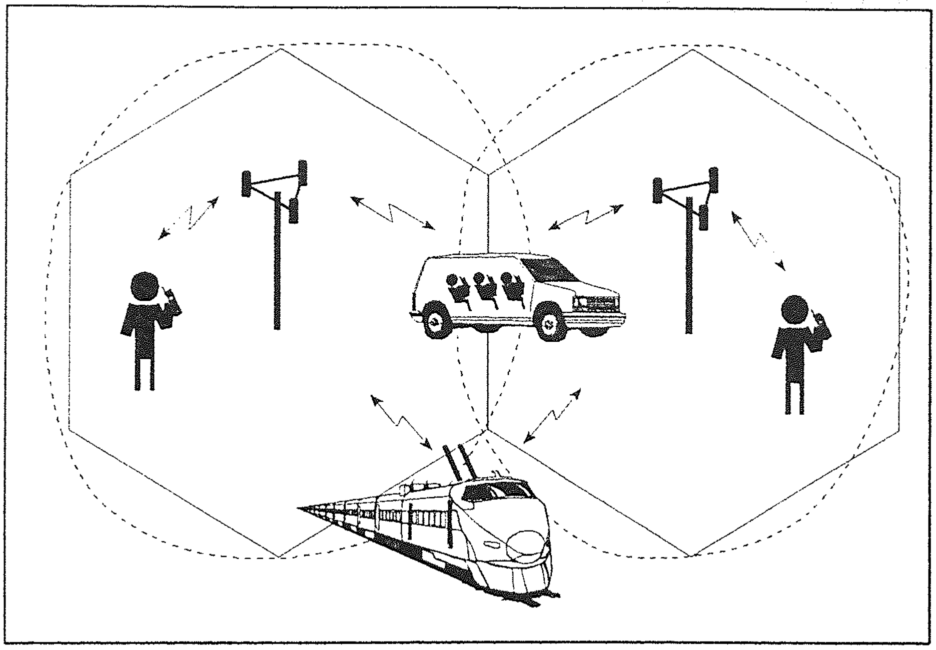
Figure (2.2): The Corner-effect in microcellular environments.

2.6.2 Multiple Call Handover

In IMT-2000, the small cell sizes anticipated will result in an increase in the handover rate compared to the previous generation systems; there is also an increase in the number of mobile terminals used for different types of teletraffic demand. Some mobile platforms may support multiple calls simultaneously. Examples include buses, taxis, trains, aircraft and boats where a number of mobile terminals are in communication with the network, as shown in Figure.(2.3). A cell boundary crossing by such a vehicle can concurrently generate multiple handover attempts, thus increasing the signalling load imposed on both the radio and the fixed network parts of the system.

Rappaport [37] developed a model which reserves a number of channels in each cell specifically for handover procedure. This method would prove beneficial when there is low to moderate loading on the system. Once the system is heavily occupied and all the traffic channels are engaged, then even achieving a successful handover might not guarantee the continuation of the call, due to the lack of traffic channels in the cell concerned. Moreover, there will be a higher rate of call blocking probability for calls initiating within that cell. Purzynski *et al* [38] implemented a prioritising scheme, similar to that mentioned in section 2.6.1, in which there was an overlap between the adjacent cells in the coverage area. Conceptually a cell was partitioned into two zones. Specifically, in one zone, the cell's own base station is able to provide service, denoted as an "inner zone". In the other, called the "transition zone", at least two base stations can establish a platform with a link of acceptable quality. It was suggested in [38] that the mobility of a platform can be described by defining a dwell time in an inner zone and a dwell time in the transition zone. The dwell time in an inner zone is the time a platform remains in communication range of only one base station. As the mobile platforms enter a new cell, handover requests are placed in a queue, awaiting the release of sufficient number of channels to satisfy their demands. Results obtained showed that handover queuing allowed significant improvement in forced termination probability, in comparison to performance obtained by using a channel reservation scheme for handovers only.

An environment where mobile platforms can support multiple independent calls represents a magnified scale of the problems facing handover for individual terminals as already discussed in section 2.6.1. Any improvement will be limited by the availability of spectral and network resources to prevent forced termination.



Figure(2.3): Simultaneous handover requests generated by multi-terminal platforms.

2.6.3 Multi-layered Systems

The deployment of a multi-tier system with macrocells overlying microcells offers system providers new opportunity to increase the networks capacity [7,6,39]. Intelligent use of the two tiers can lead to increased end-user performance and system capacity. For example, stationary users can be assigned to microcells so that they operate at reduced power and cause significantly less interference. When the microcellular capacity is exhausted, the overflow traffic can be assigned to the macrocells.

Microcell/macrocell system proposals often assign mobile terminals to a particular level according to the speed at which they are travelling. Fast moving terminals are generally encouraged to handover to a macrocell, and slow terminals typically join microcells, as illustrated in Figure.(2.4). Mobile terminals are then instructed to move between the microcellular and macrocellular planes based on their speed. This jointly reduces the handovers and increases the total system capacity [40,41,42]. Benveniste [43] suggested that calls should be initiated via the macrocell channels. A user must remain in a macrocell for a sufficiently long time before being handed over to the microcell layer. An extension to this technique estimates a mobile terminal's speed at a given point, prior to entering the microcell. This allows the system to assign mobile terminals fast enough after they cross into the cell. Thus, in all cases sufficiently slow terminals are more likely to be assigned a microcell than are fast-moving terminals.

One major motivation for using microcell/macrocell overlays is increased capacity without increased handover rates, although Eriksson *et al* [44] indicated that in some cases, system capacity could be affected to a great degree by the multiple access technique employed. A system with Code Division Multiple Access (CDMA) in both the macrocells and microcells using the same Radio Frequency (RF) spectrum, overcomes interference from the higher tier by forcing users in the microcell to transmit at a higher power than that necessary to overcome the propagation loss. The end result is that some of the cells are not overlaid; the microcell punches a hole in the macrocell. Thus no reduction in the rate of handover is achieved.

In addition to spectral efficiency, implementing a mixed cell system would protect handovers in situations like street-corner effect, where an umbrella macrocell could act as a back-up cell. In such a system, sufficient channels have to be assigned to the macrocells. It must be noted that under a low call arrival rate, a macrocell system may still perform better than a mixed cell system. This is because a higher handover failure rate will occur in the microcell system, if the same handover criteria were used for both the macrocells and the microcells. However, under a high call arrival rate, the situation is reversed. In this case the microcells clearly offer extra capacity to the system in areas with high traffic loading.

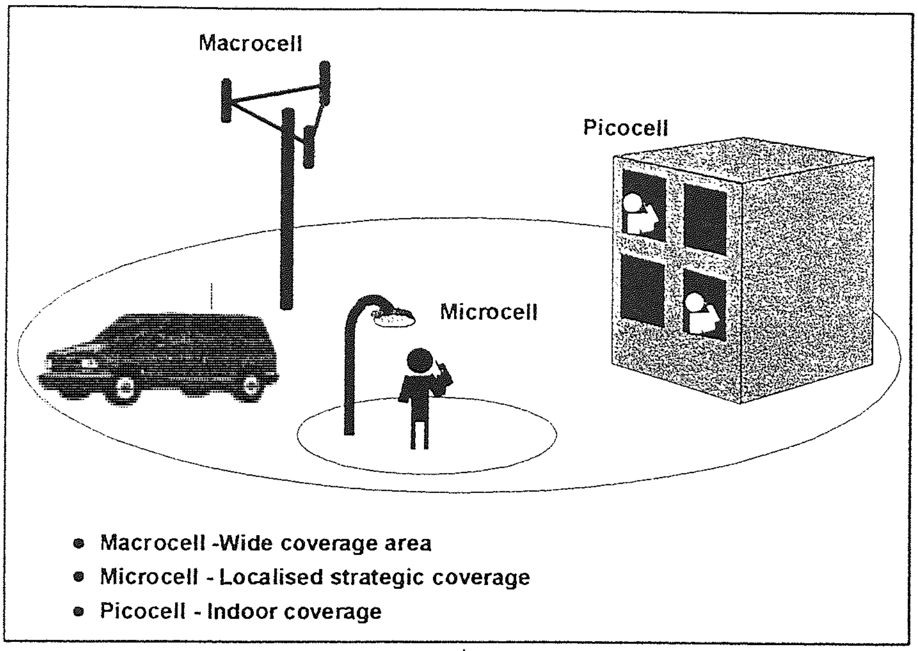


Figure (2.4): Mixed cell architecture.

2.7 Integrated Systems

The integration between terrestrial cellular networks and satellite systems is a very promising issue, since it permits the immediate provision of radio services offered by the cellular networks to areas lacking in terrestrial facilities. The satellite and the cellular networks have complementary characteristics; the main advantages of using satellites are: the expansion of the service area, and the availability of additional capacity. On the other hand, the satellite resources cannot support the same traffic volumes as a cellular network [45,46]. Therefore the satellite system acts as an extension of the terrestrial network. Researchers [47,48] regard the basic requirement of full integration to be that the insertion of the satellite system should not entail any modification of the existing GSM network (neither in the software, nor in the hardware). Although modifications to the GSM protocols will be required primarily at the mobile terminal level, a number of changes are considered necessary in some aspects of the access procedure. In an integrated system a suitable terminal (i.e. dual mode terminal) is expected to establish a call on both the RF cellular link and on the satellite link, the network features should be designed to take the maximum advantage from the presence of the two sub-systems, satellite and cellular.

Priscoli *et al* [47] considered full integration to be that which entails the reuse, as far as possible, of the GSM equipment and protocols (i.e. the same hardware and software) for the satellite system. In particular, the satellite system reuses the same channels, access technique, modulation and coding as the GSM network. This work showed that an integrated system can only work if a rural environment is considered, since different environments (i.e. urban or suburban) would require very high figures of shadowing margin (greater than 10 dB) and, consequently, too high a power level of the mobile terminal. Hu *et al* [25] considered a personal system with multiple hierarchical cellular overlays. The system included a terrestrial segment and a space segment, where the terrestrial segment consisted of microcells and overlying macrocells. At the highest hierarchical level, communication satellites comprise a space segment. The satellite beams overlay clusters of terrestrial macrocells and provide primary access for satellite-only subscribers, as shown in Figure.(2.5). Call attempts for cellular/satellite dual subscribers are first directed to the terrestrial cellular network, with satellites providing necessary overlay. At each level of the hierarchy, handover calls are given priority access to the channels.

For the system which is based on non-geostationary (mobile) satellites [25], handover may occur due to the motion of mobile terminal platforms or due to the motion of satellites. The system had suggested that once a call is served at a certain hierarchical level, it will continue service at the same or higher level but will not revert to service at a lower level. Hence, the system operation tends to create a traffic distribution in which high mobility platforms are more likely to be served by the larger cells, i.e. cells that are higher in the system hierarchy, and they will stay there until the end of the call. In effect, this will inefficiently over-utilise the channels assigned to the larger cells as mobile terminals will still be served even if they have slowed down or become stationary.

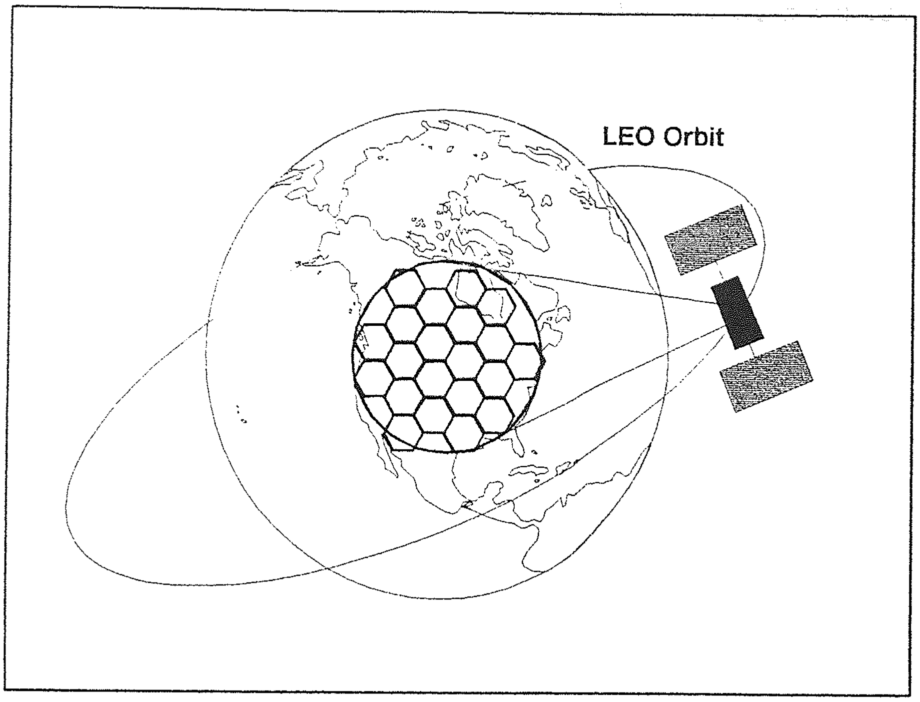


Figure (2.5): The integrated system where satellite cells overlay terrestrial cells.

2.7.1 Selection and Re-Selection in the Integrated System

Since the satellite carriers are the most valuable resource as compared to the terrestrial network channels, different studies [47,49,48,50] indicated that the GSM and the satellite carriers should be given first and second priority; respectively. The satellite spot beams can be considered as macrocells and the mobile terminal can perform the selection and re-selection procedures, considering the satellite frequency band in addition to the cellular one. At the start of a selection procedure the mobile terminal operates in the GSM system, searching for a suitable cell in every allowed terrestrial Public Land Mobile Network (PLMN). If the search is unsuccessful, a similar procedure is activated on the satellite frequency bands.

The handover from a GSM cell to a spot beam could permit a call to be continued when the mobile terminal is leaving the terrestrial cellular coverage, and still inside the spot. The handover from a spot to a cell is an interesting traffic management feature because it allows the reduction in the occupancy of the satellite resources. In this case, the mobile switching centres (MSCs) should be able to perform an inter-

MSC handover with all their cells. Apart from this problem, the handover could be carried out according to the GSM procedures [48].

2.7.2 The Mobile Satellite System

Different solutions for achieving a global coverage Mobile Satellite System (MSS) are under investigation. Promising alternative approaches seem to be those resorting to the use of Geostationary Earth Orbiting (GEO) satellites or Low Earth Orbiting satellites (LEO) [51]. In the context of personal communications, the large propagation delay and the requirements for satellite Effective Isotropic Radiated Power (EIRP) are serious drawbacks for GEO satellites when compared to LEO satellites. This explains the growing interest in LEO satellite systems for world-wide communication services, ranging from low speed data to voice communications [52].

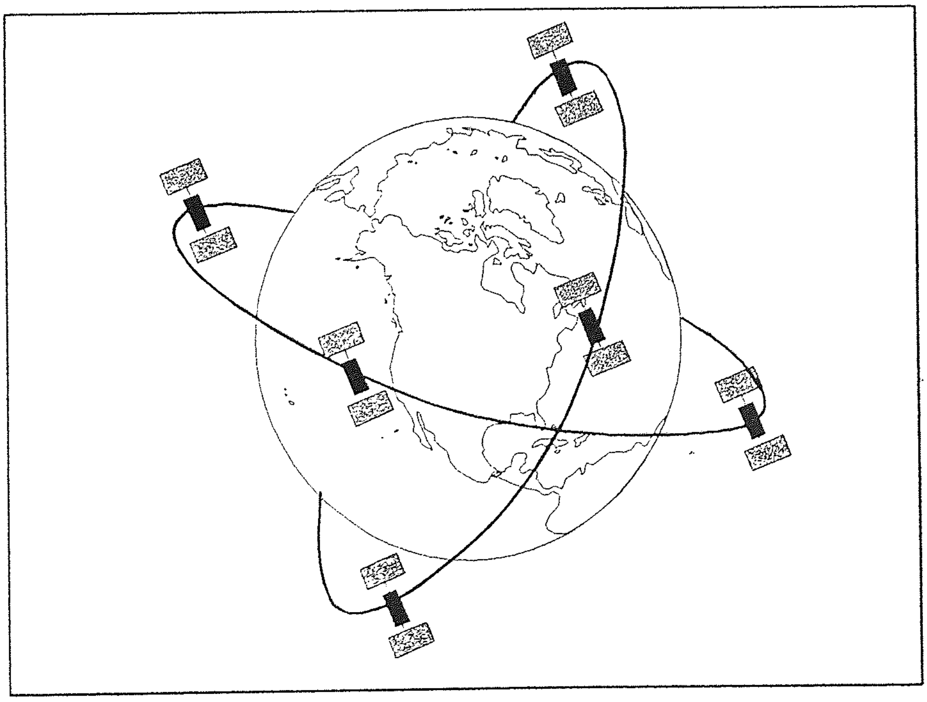


Figure (2.6): LEO satellites orbiting around the Earth

GEO satellites are fixed with respect to a terrestrial observer and they are on an equatorial circular orbit at about an altitude of 36,000 km. Only three GEO satellites are required to serve all the earth. On the contrary, LEO satellites are placed in a circular or elliptical orbit. Orbiting at an altitude ranging from 500 to 2000 km. Many LEO satellites are needed to guarantee the coverage of the whole earth at all times (see Figure.(2.6)). Despite this drawback, LEO satellites offer an advantage over GEO by using satellites at relatively low altitudes. Therefore it is less difficult to satisfy the link budget with low-power hand-held terminals as well as to reduce propagation delays and cell size.

When defining the capacity in terms of satellite circuits, the network planner has to take into account the handover traffic. Unfortunately, in a LEO satellite communication network (where handover is most often due to the motion of a LEO satellite with respect to a fixed point on the earth) the orbital speed of the satellite increases when the altitude decreases, consequently resulting in higher number of handovers. Dosiere *et al* [49] proposed a model for the handover in LEO satellite networks, where satellites were considered as single beam satellites, so that a cell is defined as the single beam coverage of the satellite and moves along with it. The model showed that the rate of handovers decreases as the altitude of the mobile satellite increases, since it will be covering a larger area with the same beam spot.

Hence, handover traffic models for terrestrial cellular networks cannot be fully used and specific models must be developed, since the handover process in mobile satellite systems becomes a two dimensional issue, involving the motion of the mobile terminal as well as the spot beam cell, as shown in Figure.(2.7).

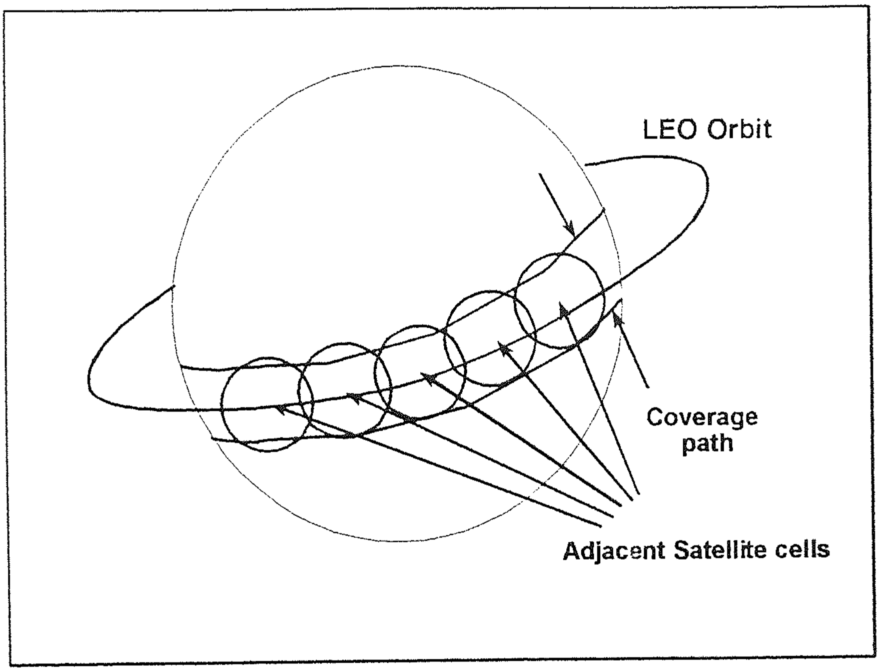


Figure (2.7): Handover due to the motion of the satellite.

2.8 Conclusion

Handover is a feature of high importance in mobile telecommunication systems. It ensures the continuity of the calls while users are moving between cells or networks. Handover should be fast enough, while at the same time should guarantee the proper call continuation. The review presented in this chapter concludes that handover performance is dependent to a great extent on the spectral capacity and the channel allocation in the system. Also, algorithms implemented assumes a satisfactory level of network congestion. An issue which has to be considered carefully when analysing maximum loading conditions. In future cellular systems, the small cell sizes anticipated will result in an increase of handover rates, thus increasing the signalling load imposed on both the radio and the fixed network parts of the system.

The increasing interest in hierarchical cellular system structure, with macrocell overlays, and the development of IMT-2000 has motivated the use of satellites for large seamless coverage. Some technical issues behind the choice of non-GEO satellites for PCS satellite constellations are transmission delay, path loss, and the antenna size required to produce narrow beams (i.e. small cells) [25]. An integration of terrestrial and satellite cellular systems requires robust handover management techniques to ensure optimum utilisation of the integrated system resources.

CHAPTER THREE

MULTIPLE ACCESS TECHNIQUES

3.1 Introduction

An important issue when designing and standardising cellular mobile radio system is the selection of the multiple access scheme, that is, the method of accommodating a number of mobile terminals signalling on one channel and separating these signals at the base station. The design of the air interface largely influences the total system design, including the fixed network, and determines the cost and operational qualities of the system.

Multiple access can be organised according to the well-known basic multiple access principles. Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA), and Code Division Multiple Access (CDMA), with practical access schemes being usually hybrids (i.e. combinations of the basic multiple access principles). A considerable amount of research work has been carried out to determine multiple access schemes appropriate for third-generation cellular mobile radio systems [4]. This is a major challenge because a large variety of scenarios and services have to be taken into account and because it is likely that a unique optimum solution does not exist.

Another major factor that determines reliable operation of the cellular system is the efficient re-use of the scarce radio spectrum allocated to the system, i.e. the channel allocation scheme. Channels can be allocated on a fixed or dynamic pattern depending on the estimated offered traffic. The basic prohibiting factor in radio spectrum re-use is interference caused by the environment or other mobile users. Efficient use of radio spectrum will also enable a reduction in the number of base stations required to service a certain geographical area and therefore a reduction in the cost-of-service.

This chapter discusses different multiple access techniques, including the Packet Reservation Multiple Access (PRMA) which is a variant of the TDMA access scheme. Particular attention is paid to the PRMA access scheme because it will be considered in subsequent chapters as the basis for the experimental work carried out in this research. This is followed by a summary of the channel allocation schemes adopted in mobile cellular systems and their suitability for different conditions.

3.2 Frequency Division Multiple Access (FDMA)

In frequency division multiple access scheme , the spectrum available is subdivided into contiguous frequency bands and the bands assigned to the mobile terminals. For frequency division duplex (FDD) transmissions using FDMA, there is a group of contiguous sub-bands occupying a predefined bandwidth for downlink transmission and a similar group of sub-bands for the uplink. A band of frequencies separates the two groups. Interference in adjacent sub-bands due to imperfect channel filtering is designed to be below an acceptable threshold. Each mobile station is allocated a sub-band, in both uplink and downlink bands. Most of the first generation analogue cellular mobile radio system use FDMA/FDD, with the speech conveyed by analogue Frequency Modulation (FM), while the control is performed digitally with the data transmitted via Frequency Shift Keying (FSK) [53].

3.3 Time Division Multiple Access (TDMA)

Time division multiple access (TDMA) is a method to enable a number of mobile terminals to access a certain channel but for only a fraction of the time and on a periodic basis. Instead of requiring *N* radio carriers to convey the communications of *N* mobile users as in FDMA, only one carrier is required in TDMA. Each user gains access to the carrier for $1/N$ of the time and in an ordered sequence.

In TDMA/FDD there are two frequency bands; one for transmission and the other for reception. When the base station transmits, the mobile terminals are switched to reception, and when the mobile terminals transmit the base station receives their signals. This significantly simplifies transceiver design compared to FDMA, where transmission and reception are performed simultaneously. Further, the base station needs only one transceiver to accommodate its N TDMA mobile users, as it only processes them one at a time, at instants according to their slot position. Another feature of the TDMA is the transceiver's ability to monitor other channels during slots when it is neither transmitting nor receiving signals.

3.4 Code Division Multiple Access (CDMA)

Originally known as Spread Spectrum Multiple Access (SSMA), this is another method that allows multiple users to access the mobile radio network. Each user is allowed to use all the bandwidth, like TDMA, and for the complete duration of the call, like FDMA.

In CDMA all users communicate over the same bandwidth using an individual Pseudo-random Noise (PN) sequence to spread the modulated data signal over a wider bandwidth. The users' transmission are therefore stacked upon one another in the frequency domain causing interference to one another, yet due to knowledge held in the coding, a receiver may demodulate the required signal once the codes are synchronised [54].

Direct Sequence (DS-SSMA) or CDMA, is one of two basic types of spread spectrum techniques [55]. The second is called Frequency Hopping (FH-SSMA). The latter scheme was proposed as an alternative to the FDMA in the first generation cellular systems [56].

3.5 Packet Reservation Multiple Access (PRMA)

Packet Reservation Multiple Access (PRMA), is a statistical multiplexer of speech for wireless communication systems. It exploits the silence periods in speech to accommodate more users than there are channels available. A speech activity detector classifies speech into talkspurts and silence periods. PRMA is similar to TDMA in the sense that the transmission time scale is based on frames, each containing a fixed number of time slots. However, unlike TDMA, where the terminals are granted access to a channel for the entire duration of a call, PRMA voice terminals are granted access to the channel only during talk spurts.

Each packet is composed of a header containing signalling information and an information field. At the beginning of each activity period a user contends for a time slot. If the base station is able to decode the header of the first packet of the talkspurt, the contention is successful and the slot is "reserved" for that user for the whole duration of its talkspurt. However, if contention is unsuccessful, in order to off-set the re-transmission of the contending users, access is attempted according to a re-transmission or permission probability, permissions at different terminals being statistically independent.

Congestion leads to access delays and, since speech packets require prompt delivery, those that are delayed beyond a certain time are discarded or dropped. Dropped packets affect speech quality and hence the packet dropping probability is an important measure of PRMA performance. Speech clipping at the beginning of a talkspurt proved not to be very disruptive. Subjective tests point out that speech quality is preserved until packet dropping probability is above 0.01 [57,58]. When a terminal is in the contention phase, collisions may occur with contending users in the same cell, and with active users in co-channel cells. The overall effect will depend on the capability of the base station receivers for recognising one out of many contending users. This may be generally accomplished by introducing a *capture* mechanism [59], which is based on the capability of the base station to recover the packet header even when corrupted, up to a certain degree, by interference.

Once reservation has been gained by a given user, signals from co-channel cells may produce interference on the talkspurt in progress. Different reservation policies have been proposed. Frullone *et al* [57] suggested that the header and the information field can be characterised by different coding protection levels, so that the carrier-to-interference ratio required for the information bits (CIR_{info}), can be different from the carrier-to-interference ratio required for the header bits (CIR_{head}). The slot reservation could therefore be kept or lost according to pre-set CIR_{info} and/or CIR_{head} thresholds. In other words, the criterion will be either a corrupted packet header or a corrupted information field in the packet. Goodman *et al* [60] reported that PRMA performance is sensitive to the choice of the transmission probability. If transmission probability is too low, terminals wait too long between successive transmissions and eventually drop packets as the waiting time exceeds the system limit. If transmission probability is too high, excessive collisions occur, and the resultant congestion will lead to high packet dropping probability.

As with the case of TDMA, co-channel interference probability increases as cellular cluster size decreases. Comparing the performances of PRMA with those of dedicated channel multiple access (TDMA) schemes using the RSSI criterion, it is evident that in equivalent situations (i.e. the same cluster size) the interference probability of PRMA is always lower than the corresponding probability of dedicated channel multiple access schemes [61]. This effect can be easily explained considering that, in the PRMA case, only active talkspurts are transmitted and the interference level is correspondingly reduced. Furthermore, the interference probability of systems with dedicated channels is increased by the hysteresis introduced in the handover procedure, which forces mobile terminals to operate with very low values of the received signal level.

3.5.1 Voice and Data Integration over PRMA

A flexible medium access scheme is highly desirable for Third Generation cellular networks because of the expected wide range and variety of teletraffic demands. A packet access protocol is well suited to such tasks since it can easily offer variable bit rates and also carry bursty traffic. Data-only and combined data-voice traffic is expected to increase, and will comprise a substantial amount of the carried traffic. A crucial issue regarding the integration of voice and data in PRMA is the contention, or permission probability. Considering a system in which both voice and data terminals contend for all free slots, permission probability for data access should be lower than that for voice access. This will ensure higher priority for the delay sensitive voice packets.

Lee *et al* [62] proposed a data multiple access protocol that combines random access (i.e. slotted ALOHA) with a slot reservation mechanism (burst mode PRMA). It aims to statistically multiplex data packets transmitted by data terminals with speech packets transmitted by voice terminals over the shared uplink channel. Voice terminals were assumed to be using the access protocol in Extended TDMA (E-TDMA) [3], which is an enhancement of the emerging North American digital cellular standard (IS-54) [63]. By treating voice slot assignments with pre-emptive priority over data, data traffic becomes transparent to the voice terminals. Furthermore, the proposed protocol can accommodate multi-slot assignments per frame to match the throughput needs of individual data terminals, as well as application dependent data transmission priorities.

Dunlop [67] considered a hybrid packet system with a fixed boundary between voice and data sources, and a hybrid packet system with a dynamic boundary between voice and data sources. In each case, PRMA in burst mode is used for voice talkspurts and PRMA in random mode (slotted ALOHA) is used for data sources. In the fixed boundary structure the position of the boundary is fixed according to the anticipated mix of voice and data services and also the acceptable delay requirements for data and the acceptable packet loss for voice.

Reported results showed that standard PRMA provided the best delay response for data services followed by dynamic boundary scheme and fixed boundary, respectively. It also showed that at high data loads PRMA produces the highest voice packet loss, while at low and moderate loads, voice packet loss was still below the 1% threshold. Dunlop [67] had also evaluated a packet access mechanism for integrated voice and data transmission using random access (slotted ALOHA) as compared to the reserved access (burst mode PRMA). For data transmission, there was no packet dropping threshold, resulting in a superior access delay performance as compared to conventional PRMA. This was a consequence of the much shorter collision resolution interval experienced with the random access mechanism. This is significant in the case of data transmission because colliding packets are continually offered for transmission until successfully received by the base station. On the other hand, the performance of PRMA for voice exceeded that of the random access mechanism and the performance gain increased as the bit rate increased.

Narasimhan *et al* [68] suggested a scheme in which the voice and data sub-systems are logically separated. The total available bandwidth is divided into three regions: voice information, voice contention, and data regions. The bandwidth available for the data users depends on the fulfilment of the QoS requirements for voice users, which is assumed to be a limit on the number of dropped voice packets. Numerical results indicates that a significant amount of data traffic can be supported without sacrificing the voice capacity of the system.

It can be concluded that service integration can result in high bandwidth utilisation through statistical multiplexing of connections belonging to service classes with different source activity factors, particularly voice sources with an activity factor equal to 0.44 [69]. The utilisation becomes even higher when more bursty connections, such as data, are multiplexed on the same link. Also by offering two alternative classes for voice calls, the system can control its blocking and access delay performance.

3.5.2 Alternative Packet Access Techniques

Research findings have shown a number of new mechanisms, in principle based on PRMA, but aimed at exploring this technique to find better ways of improving its performance [64,65,66]. Some were aimed at integrating voice and data transmission by taking advantage of the statistical multiplexing inherent in the PRMA scheme, as have been discussed in the previous section. Others have concentrated on introducing new reservation policies, these are described in the following section.

Since permission probability in PRMA is introduced in order to offset subsequent attempts after a collision had occurred, Frullone *et al* [57] questioned whether or not this mechanism should be actively operated on the first attempt (i.e. at the beginning of each talkspurt). An alternative strategy was envisaged in which the first attempt to obtain the reservation doesn't require a permission to transmit. A random value of permission would be generated only if a collision occurred. With permission probability equal to one for new talkspurts; it was shown that the new strategy improves the performance for low traffic loads only, while it is the same as PRMA for high traffic loads.

Frullone *et al* [57] had also found that the dropping probability changes greatly upon varying carrier-to-interference ratio in the packet header, whereas the information field carrier-to-interference ratio is kept constant. The dropping probability was found to increase as the header protection decreases (the higher the required CIR_{head} the higher is the header protection) while interference probability exhibits an opposite behaviour. This can easily be explained considering that a higher header protection reduces the rate of reservation losses due to interference (and thus the re-transmission attempts), thus forcing the terminal to continue its transmission even when the information field is corrupted by interference.

Wong *et al* [70] proposed a multiple access scheme, Shared Time-Division Duplexing (STDD), suggesting that both uplink and downlink traffic can share a common control channel, thereby achieving high statistical multiplexing even with low population of simultaneous conversations. The approach of having separate control and speech channels was reported to facilitate a better grasp of the overhead efficiency problem, which is common in packet switched networks.

In order to increase the probability of successful reservation, Tafazolli *et al* [71] modified the PRMA scheme by subdividing an unreserved slot into several sub-slots, and in each one only a reservation packet, which carries a small amount of information, such as terminal ID, was transmitted. All terminals which have information to transmit send a reservation packet in a sub-slot with a permission probability. If more than one terminal has successful transmission, the base station will always allocate the slot to a contending voice terminal. Voice traffic was given higher priority over the data traffic to minimise the speech transmission delay and hence the average packet dropping probability. Although this protocol allows more access to the channel by securing reservation slots, it suffers from degrading speech quality as the number of mobile stations waiting with transmissions becomes more than the available uplink slots, causing packets to suffer too much delay and to be dropped.

A second protocol whereby a hybrid scheme using circuit switching for voice users and packet switching schemes for data users was also proposed [72]. A voice user gives up a fraction of its slot after it has entered the silent period, and uses the rest of its slot for signalling purposes. This sub-slot can be used to request back its slot as soon as it enters a talkspurt state. Data users do not have to contend for any slot, since the base station has pre-knowledge of the subscribers which have traffic to transmit. As mentioned above this information is obtained through the call set-up procedure. Once a slot is released, the base station will broadcast the allotment of this slot in the next frame. The protocol avoids wasting time slots due to repetitive collisions.

It is important to mention here that this scheme could work effectively for users who wish to transmit data while they are transmitting voice, because the terminal will be using the reserved time slot efficiently in silence periods, provided that there are no constraints on the type of data to be transmitted. If the time slot was used just for signalling during silent periods, then it will be very much like a circuit switched scheme, as the slot would be reserved for the duration of the call.

The two packet access mechanisms considered for Advanced TDMA (ATDMA) [73,74], are both developments of PRMA and are known as Block Reservation Multiple Access (BRMA) [75] and PRMA++ [76]. The essential difference between these mechanisms is that in BRMA all slots on the uplink may be used for resource reservation, whereas in PRMA++ the slots on the uplink are separated into reservation slots (R slots) and traffic information slots (I slots). In the case of BRMA, reservations are acknowledged on a paired down slot, and in PRMA++ special slots (A slots) are reserved on the downlink for acknowledgement purposes and uplink slot allocation. Both mechanisms were evaluated for voice applications only.

With particular interest in the optimum choice of the number of reservation slots in the uplink, results obtained in [75] show that if the number of R slots is too low it is expected that collision resolution process will be the major contribution to access delay and packet loss (in the case of voice). If the number of R slots is too high then this will reduce the number of collisions due to contention but will, at the same time, decrease the number of I slots per frame. This will produce longer queues for I slot allocation and increased access delay and voice packet loss from this cause. As ATDMA is specifically concerned with the concepts of an adaptive air interface capable of accommodating variable bit rate services, the initial conclusion was that PRMA++ is likely to provide a greater degree of flexibility than BRMA in such an environment.

3.6 Channel Allocation Schemes

For a given spectrum and a specific technology used, the traffic-carrying capacity of a cellular system depends on how the channels are managed. The following sections describe the types of channel allocation schemes used or that have been suggested for use in the cellular networks:

3.6.1 Fixed Channel Allocation (FCA)

Fixed channel allocation (FCA) is used where each cell is permanently assigned with a fixed set of nominal channels. A definite relationship is assumed between each channel and each cell in accordance with co-channel interference constraints [77]. If a new call finds that no free nominal channel is available, the call is blocked. The current GSM system utilises FCA.

FCA can be done following a uniform allocation, i.e. where the same number of channels is allocated to each cell. This uniform channel distribution is efficient if the traffic distribution of the system is also uniform. Also channels can be allocated in a non-uniform pattern, depending on the expected traffic loading for the different cells of the system. These are referred to as Channel Borrowing Schemes. In a channel borrowing scheme, a cell that has used all its nominal channels can borrow free channels from neighbouring cells to accommodate new calls. A channel can be borrowed by a cell if that channel does not interfere with existing calls. When a channel is borrowed, several other cells are prohibited from using it. This is called *channel locking*. The number of such cells depends on the cell layout and the type of initial allocation of channels to cells.

The borrowing strategy can be Simple Borrowing [78], whereby any channel in any cell can be borrowed subject to interference requirements only. Alternatively a Hybrid Borrowing [77,79,80] strategy can be used in which the set of channels assigned to each cell is divided into two subsets: one is nominally for use inside that cell, and the other one is allowed to be lent to neighbouring cells. The ratio between the two sets is dependent on the estimation of traffic conditions and can be adapted dynamically in a scheduled or predictive manner.

For heavy traffic loading, the performance of borrowing schemes could deteriorate as compared to uniform FCA because of channel locking. The channel usage efficiency would drop drastically, causing an increase in blocking probability and a decrease in channel utilisation [81]. However, the performance of the non-uniform FCA scheme is superior to the uniform FCA in light and moderate traffic conditions only, because it allows the system to respond to traffic fluctuations.

3.6.2 Dynamic Channel Allocation (DCA)

In contrast to FCA, there is no definite relationship between the cells of the system and the channels that are used in them. In dynamic channel allocation (DCA) all channels are kept in a central pool and are assigned dynamically to radio cells as new calls arrive in the system. Any cell can use any channel as long as the interference constraints are satisfied [82]. Once a call is completed, its channel is returned to the central pool.

In DCA, a channel is selected according to certain interference constraints. Also the cost of using each candidate channel is evaluated. The selected cost function might depend on the future blocking probability in the vicinity of the cell concerned the usage frequency of the candidate channel, the reuse distance, channel occupancy distribution under current traffic conditions, radio channel measurements of individual mobile terminals, or the average blocking probability of the system [81]. DCA can be performed by a centralised controller, whereby the specific cost function used is the determinant factor. Such a system is characterised by high centralisation over-head [83,84]. In contrast, Distributed DCA schemes use either local information about the current available channels in the cell's vicinity (cell-based) or signal strength measurements [85,86].

In the cell-based schemes, a channel is allocated to a call by the base station at which the call is initiated. The difference with the centralised approach is that each base station keeps information about the current available channels in its vicinity. The channel pattern information is updated by exchanging status information between base stations. The cell-based scheme provides near-optimum channel allocation at the expense of excessive exchange of status information between base stations, especially under heavy traffic load.

On the other hand, the signal strength measurement schemes use only local information at a particular base station, without the need to communicate with any other base station in the network. Thus, the system is self-organising and channels can be placed or added any where, as needed, to increase capacity or to improve radio coverage in a distributed fashion. These schemes allow fast real-time processing and maximal channel packing at the expense of increased co-channel interference probability with respect to ongoing calls in adjacent cells, which may lead to undesirable effects such as interruption, deadlock, and instability.

3.6.3 Scheme Comparison

In FCA, the allocation control is made independently in each cell by selecting a vacant channel among those allocated to that cell in advance. In DCA, the knowledge of occupied channels in other cells as well as in the cell in question is necessary. The amount of control is different in each DCA strategy. If the DCA requires a lot of processing and complete knowledge of the state of the entire system, the call set-up delay would be significantly long without high-speed computing and signalling. As discussed in [87], the implementation complexity of the DCA requires a great deal of processing power to determine optimal allocations and a heavy signalling load. On the other hand, FCA requires a complex and labour-intensive frequency planning effort to set up a system, which is not the case for the DCA schemes.

Regarding type of control, FCA is suitable for a centralised control system, while DCA is applicable to a decentralised control system. A centralised control system creates a huge control volume in a microcellular system, which can lead to bottleneck. One solution is to divide the control area into several sub-areas of suitable size.

3.7 Conclusion

Basic PRMA is a scheduled access packet transmission technique which operates in a reservation mode for voice traffic. It uses a contention mode for obtaining a reservation for a voice terminal during periods of activity. A random access mode was defined for data traffic (no reservation) but in a cellular mobile environment it is most unlikely that the random access mode would be used for individual (independent) packets. What is clear, however, is that the transmission of voice and data in blocks, corresponding to the periods of activity, offers the potential of occupying valuable radio resource only during these periods, thereby offering potential for increased efficiency over fixed allocation TDMA systems. This has lead to the investigation of algorithmic changes for adapting PRMA to the needs of cellular communications. Service integration can result in high bandwidth utilisation through statistical multiplexing of connections belonging to service classes with different source activity factors.

In microcellular environments, the mobile terminal's high mobility will temporarily cause high spatial traffic inhomogenies that FCA schemes are not able to cope with, even if a frequency pre-planning could be afforded. That is the reason why the introduction of microcells implicitly requires DCA schemes which are able to adapt to such variations. For the same blocking rate DCA has a lower forced termination rate than FCA. In FCA a call must be handed over into another channel at every handoff because the same channel is not available in adjacent cells. In DCA the same channel can be assigned in the new cell if co-channel interference does not occur.

Generally, there is a trade-off between quality of service, the implementation complexity of the channel allocation algorithms and spectrum utilisation efficiency. Under low traffic intensity, DCA strategies perform better, due to the fact that DCA uses channels more efficiently than FCA. In the case of FCA, channels are pre-assigned to cells, so there are occasions when, due to fluctuations in traffic, calls are blocked, even though there are channels available in adjacent cells. However, FCA schemes become superior at high offered traffic, especially in the case of uniform traffic.

Although PRMA allows a reduction in interference probability, by restricting transmission to activity periods only, this is not sufficient to permit the use of a DCA scheme. The use of packet transmission in conjunction with DCA in microcellular or picocellular environments, characterised by the difficulty to identify regular reuse patterns, requires some specific technical solutions for facing the uncontrolled interference. Therefore DCA and PRMA could well co-exist in cases of non-voice communications characterised by a lower activity factor.

CHAPTER FOUR

SIMULATION AND MODELLING

4.1 Introduction

Due to rapid development and availability of computer-based technologies, there has been an increased desire to incorporate them into areas such as communication and computer networks. This has resulted in communication and computer networks becoming larger and more complex, hence the task of designing and managing these systems has become increasingly more challenging. Moreover, the validation of designs for major networks necessarily requires intricate simulation tools; manual calculations are no longer feasible to perform design validation. An automated simulation tool provides the flexibility of creating an abstract network model that can be used to perform analysis such as the assessment of alternative designs and establishing the merits and de-merits of different operational policies. Hence, a network designer can use a simulation to explore the behaviour of a proposed system (without incurring the task of constructing it), and the impact of proposed modifications to an existing system can be ascertained without disturbing the actual network.

Typically, an off-the-shelf simulation tool performs a simulation in accordance to a set of a specific range of parameters and variables that are designed to analyze certain aspects of a network's performance. However, this may impose limitations on the type of analysis that can be performed. For example, the COMNET III simulation tool is capable of simulating a range of fixed line networks with all the relevant protocols and control policies. Also it is capable of representing an air link in terms of bit rates only. However it does not provide a facility for simulating any channel access techniques, e.g. TDMA, FDMA or CDMA. This thesis used the COMNET III simulation tool and performed the task of simulation in two parts. In the first part a GSM PLMN model was simulated and a measure of the packet delays over a fixed network (the fixed part of

PLMN) was obtained. In the second part of the simulation it was necessary to develop code in order to simulate a further detailed PLMN model. Code development was necessary at this stage because the COMNET III tool could not perform simulation of channel access policies. The simulation tool was developed using C++ object oriented algorithms. This tool was used to implement a number of air interface protocols and the simulation focused on measuring the delay figures during handover.

This chapter concentrates on describing the two cellular network models and the methods used to create simulation models of them. The task of simulating these models as well as presenting the design issues and parameters for each simulation is reserved for chapter 5. However this chapter is organized as follows: section 4.2 defines the work carried out with respect to the Open System Interface (OSI) protocols. Section 4.3 details the simulation package COMNET III. This is followed by an outline of the multiple access technique modelling and the C++ algorithms used to simulate the microcellular network.

4.2 The OSI Model

The OSI model for an open communication system defines seven layers; a description of these layers is shown in Table (4.1). The signalling between all the interfaces from a GSM mobile station to the MSC takes place in the lower three layers (i.e. layer 1 to 3) [3,88].

All the schemes and mechanisms used to make communications possible on the mobile radio channel, with some measure of reliability between a mobile and its base station, are represented by the physical layer, or the Layer 1. These mechanisms include modulation, power control, coding, timing, and a host of other details that manage the establishment and maintenance of the channel. Layer 2 is the data link layer which is responsible for the correct and complete transfer of information blocks between Layer 3 entities over the GSM radio interface. This includes peer-to-peer transmission of signalling data in a defined frame format, and the establishment and termination of one or more (parallel) data links on signalling channels. The base station passes signalling messages between the mobile station and the BSC or MSC; however, it seldom takes part in the conversations except when it has to respond to commands for adjustment in its operation. Layer 3 (or the network layer) in the GSM architecture, which is also referred to as the signalling layer, uses a protocol that contains all the functions and details necessary to establish, maintain, and terminate mobile connections for all the services offered within a GSM PLMN.

Layer 7	APPLICATION	Application protocols, user-oriented provision of communication data.
Layer 6	PRESENTATION	Application specific format transfer.
Layer 5	SESSION	Connection of application processes, billing
Layer 4	TRANSPORT	Flow control for point-to-point connections.
Layer 3	NETWORK	Connection and switching of communication links.
Layer 2	DATA LINK	Control of signalling links, block transfer of signalling data.
Layer 1	PHYSICAL	Physical transmission, coding, error correction, modulation.

Table 4.1 The seven layers of the OSI model.

Three sublayers are defined for layer 3; a sublayer can be regarded as an entity in itself, which handles the tasks and functions of a specific segment of signalling in the GSM PLMN. These sublayers are:

(i) *connection management* (CM): This sublayer contains the call control entity which is responsible for establishing, maintaining, and releasing call connections for communication links;

(ii) *mobility management* (MM): This sublayer is responsible for the support of user mobility, registration, management of mobility data and checking the user and the equipment identity;

(iii) *radio resource management* (RRM): The tasks covered by the RRM sublayer are closely connected to the physical layer. Procedures within this sublayer are defined to cover tasks like channel assignment procedures, channel change and handover procedure, change of channel frequencies, hopping sequences (hopping algorithms), and frequency tables [89]. The RRM sublayer handles all the procedures necessary to establish, maintain, and release dedicated radio connections.

The work carried out in this research project deals with the signalling and time delays associated with performing a complete handover procedure. Therefore the network layer, and in particular the RRM sublayer is of specific interest. Moreover, the code programming (which is an investigation into packet access mechanisms) is aimed at exploiting layer 3 procedures to perform fast and reliable transfer of communication between different cells as the mobile terminal roams in the cellular coverage area.

4.3 COMNET III

COMNET III is a performance analysis tool that is used in the area of computer and communication networks. The analysis performed by this tool is based on a description of the network under consideration, its control algorithms and workload. The network modelling approach used in COMNET III is designed to accommodate a wide variety of network topologies and routing algorithms; these include: (1) LAN, WAN and inter-networking systems; (2) circuit, message and packet switching networks. COMNET III simulates the operation of the network and provides measures of network performance in terms of throughput and teletraffic blocking. COMNET III supports a modular approach to the construction of a network, in which the modules are objects with associated parameter lists that represent entities in the 'real-world'. Example modules are links and nodes. COMNET III has the capacity to adjust the parameters of different objects to match the various design requirements.

COMNET III is mainly dedicated to modelling fixed line networks. Radio links can only be represented in terms of bandwidth and transmission bit rate, which is not adequate for air interface analysis. Although certain error probabilities can be applied to model lossy links, the software does not include RF activity, neither does it model fading or shadowing. Hence, the simulated model assumes an error-free air link between the mobile terminals and the base stations. Details of the network parameters will be discussed in the following chapter.

4.3.1 Network Model

Figure. (4.1) shows a block diagram of a multi-tier cellular network that will be simulated with COMNET III. It consists of four micro-cellular base stations, and each pair of microcells is connected to a LAN. The LAN in turn is connected to a macrocell base station via a Signalling Point (SP). Each macrocell is connected to a BSC and both BSCs are served by the same MSC. LANs are deployed to allow the exchange of signalling traffic between each pair of microcells without resorting to the respective BSC, particularly for intra-BSC handovers where the mobile terminal is handed over between two cells within the same BSC. This technique has already been implemented by Steele *et al* [13] and has been shown to reduce congestion in the fixed network.

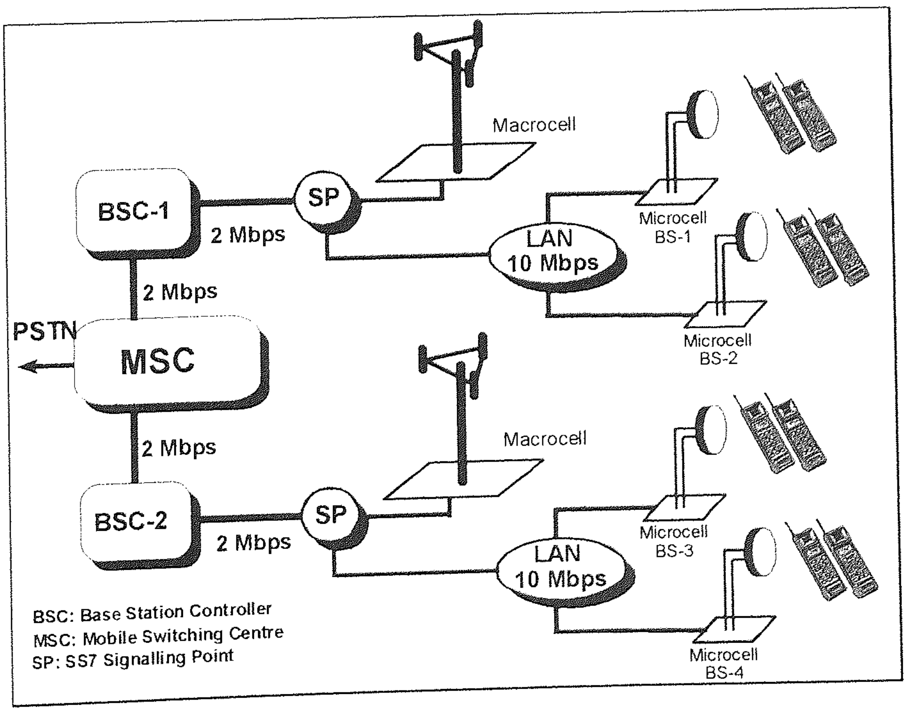


Figure.(4.1): A multi-tier cellular network

In order to avoid a complex and cumbersome model layout, each base station was allocated a single channel serving eight mobile terminals, this is the maximum number of users that can be accommodated on a GSM radio frame. To simulate a heavily loaded network, variable loading levels were applied to the links in the fixed part of the network. This allowed the flexibility of representing a maximum traffic loading scenario without facing the physical difficulty of creating and integrating every single mobile terminal. All the calls considered in this model are mobile to PSTN calls, mobile to mobile calls were not included.

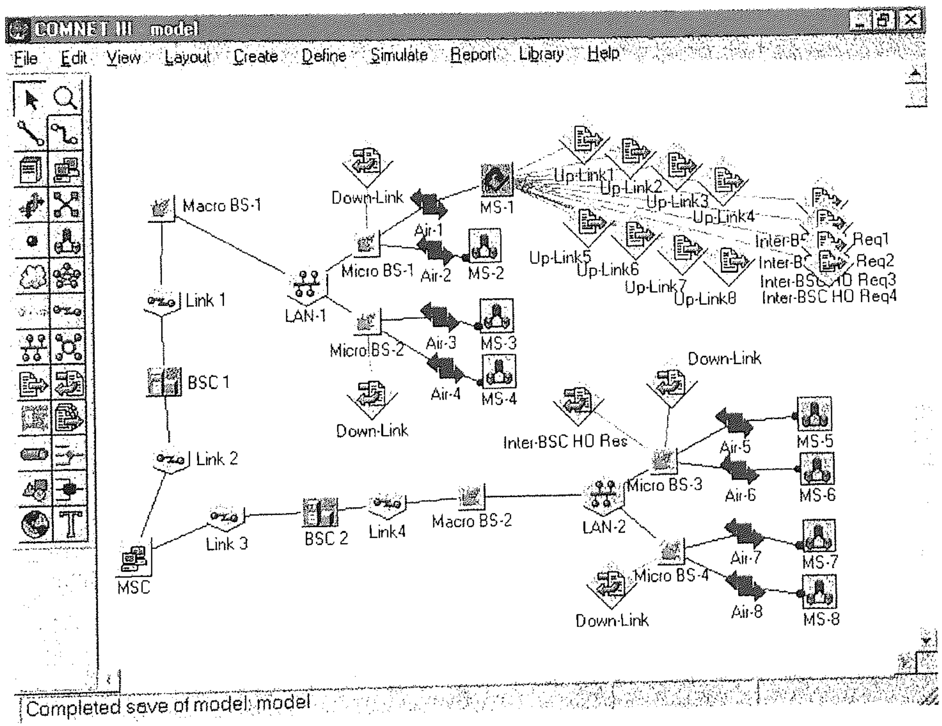


Figure.(4.2): COMNET III Layout of the GSM Public Land Mobile Network.

Figure.(4.2) shows the network layout as represented by the COMNET III graphics. Blocks MS-1 to MS-8 represent single channels each serving eight users, Up-link1 to Up-link8; respectively. In addition to the eight speech sources (Up-link1 to Up-link8), four Inter-BSC handover requests sources are used to generate handover request packets from four mobiles in each cell. Speech packets and handover packets are expected to use the same user channel. The "spider-web" figure of the MS-1 block and its eight transmission sources (as shown in Figure.(4.2)) represents the standard physical layout of elements used to model a channel. Blocks MS-2 to MS-8 are integral versions of MS-1, with exactly the same functionality.

4.4 Multiple Access Technique Modelling

In the COMNET III model, single measurements were taken at specific instants during the simulation. These were measurements of the delays encountered by the signalling packets exchanged between certain points across the PLMN. This type of model only allowed for the inspection of the fixed part within the GSM PLMN. The second part of the simulations (carried out by the simulation tool developed using C++ programming code) evaluated the performance of the microcellular system under different air access techniques and handover policies. The standard GSM handoff procedure was tested in both TDMA and PRMA access schemes. In chapter 5 the results obtained from TDMA and PRMA will be compared with those of the new improved handover strategy, the Advanced PRMA. Figure.(4.3) outlines the main entities in this simulation.

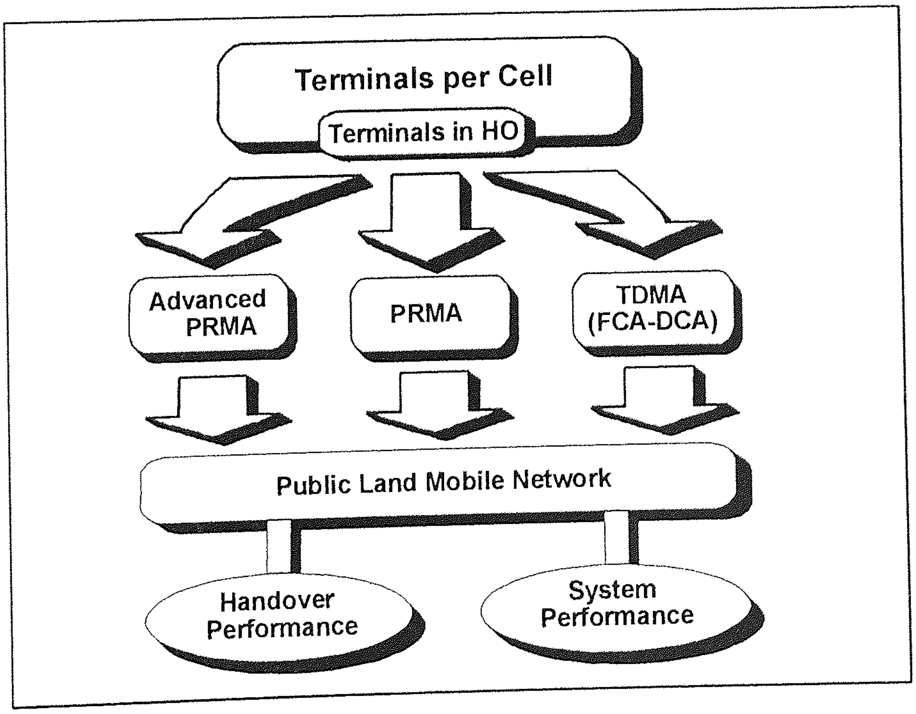


Figure.(4.3): TDMA, PRMA & Advanced PRMA modelling to investigate handover performance.

The network consisted entirely of microcells and it was our intention to examine complete handover procedures in systems with no overlying macrocells. Every mobile terminal was modelled as an independent entity within the network, i.e. in terms of generating calls and sustaining handovers. This was done so that the contribution to the overall performance of the system could be better evaluated.

The network configuration in this part of the simulations comprised two clusters of seven cells. Each cell was allocated 16 duplex channels. Each cluster was controlled by a separate BSC, both supervised by the same MSC. A number of channel allocation schemes and handover policies were implemented, such as FCA and DCA, with prioritized and non-prioritized handover. Emphasis was placed on the handover performance of mobile terminals handing over between the two clusters. Figure.(4.4) shows the cellular layout represented in this model. The arrows indicate the direction of travel for the mobile terminals handing over between the two clusters. In chapter 5 a complete description of the design and operation parameters for this network, along with a comprehensive demonstration of the results obtained, is given.

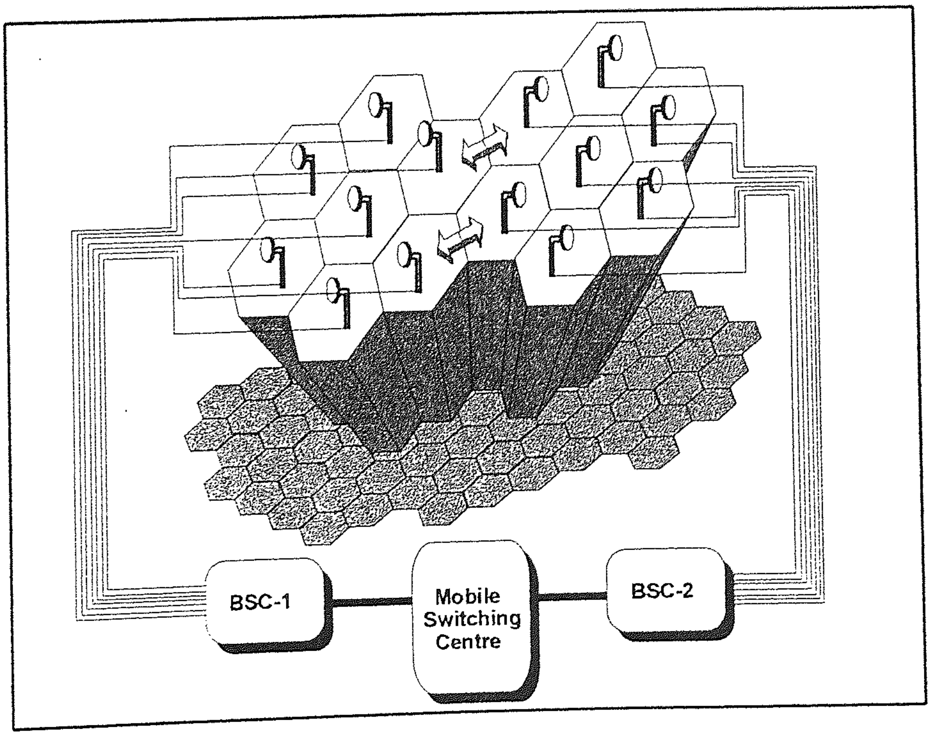


Figure.(4.4): 14 cells Micro-cellular system, with terminals crossing two pairs of adjacent cells.

4.5 Conclusion

This chapter presented the modelling methods used to build and analyze two micro-cellular systems. In particular, it focused on the delays and overall system performance during handover procedures. The first part of the simulations, a PLMN model using COMNET III, examined the behaviour of the individual speech and handover packets while transferring calls between cells of different BSCs. In chapter 5, this model will be used to provide real-time delay measurements of single events instead of complete handover procedure sequences. COMNET III is capable of allocating different levels of teletraffic loading to individual links in the network. Hence, it was possible to simulate a PLMN with maximum loading without the need to physically create and place every single mobile terminal on the COMNET III graphical user interface.

The second part of the simulations used C++ algorithms to design a model which was an extended version of the network constructed in part one. Specific multiple access schemes were implemented and different levels of teletraffic loading could be considered. However, unlike the first part, the simulations in the second part focused on complete procedures with regard to initiating and sustaining calls and handovers. Therefore, comparisons among individual terminals' performances were possible. Object oriented programming proved suitable for detailed modelling of mobile terminals because it permits the concurrent execution of a number of functions in a single environment.

CHAPTER FIVE

ADVANCED HANDOVER SIGNALLING STRATEGY

5.1 Introduction

This chapter presents the design parameters and the models used to evaluate the improved signalling strategy and compares it with existing schemes in TDMA and PRMA. It explains the reason why PRMA is considered as a medium access protocol for implementing this strategy. A crucial fact to consider is that the new handover signalling strategy concentrates on the steps and the functional units responsible for performing the handover procedure, and that it is not concerned with the criteria that initiates the handover process.

FCA and three different DCA schemes are simulated as part of the TDMA medium access protocol while PRMA++ and the Advanced PRMA are simulated for the PRMA medium access protocol. The simulation of FCA and DCA was necessary to compare the handover delay output of each scheme; it was not aimed at examining the spectral capacity of such schemes as this aspect has already been tested and evaluated by earlier research work [91,92,93]. It is noticeable that the majority of past research studies into handover performance have always accepted time delays as long as they were below the current limits for the GSM standard, and rarely considered lowering the delay thresholds below these limits. The work presented in this thesis suggested performing faster handovers within lower time limits, thereby achieving higher probability of successful handovers and better spectral efficiency.

The following sections will explain in detail the steps involved in performing a complete handover procedure in the GSM standard. This will be followed by a detailed description of the principle of the advanced handover strategy and its functional requirements. A model for the fixed part of the PLMN is simulated in section 5.4 using COMNET III software where the behaviour of individual speech and handover packets are assessed. Design parameters for the TDMA and PRMA medium access protocols are demonstrated in section 5.5; also in this section a description of the mechanism used to implement the advanced signalling technique in a PRMA medium is given. Results and performance analysis follows in section 5.6.

5.2 Handover in GSM

The GSM system distinguishes between different types of handovers. Depending on what type of borders the mobile terminal is crossing, a different entity may have to control the handover to ensure that a channel is available in the new cell.

5.2.1 Types of Handover

- *Intra-Cell Handover Procedure:* Intra-cell handover occurs between radio channels of the same base station. This capability is used when the radio channel carrying the call is subject to interference or when a radio channel or channel equipment carrying a call has to be taken out of service for maintenance or other reasons. The decision to invoke and execute the procedure is taken autonomously by the Base Station Sub-system (BSS), see Figure.(5.1). The decision process in the BSS is based on the internally available radio and resource parameters.

- *Inter-Cell Handover Procedure:* Inter-cell handover procedure can be one of two types. The first is known as *intra-BSS* handover and occurs between channels pertaining to different cells of the same BSS. In this case the procedure is dedicated and executed autonomously by the BSS. The second is known as *inter-BSS* handover, this occurs between channels pertaining to different cells of two different BSSs, in this case the procedure is performed by the MSC.

- *Inter-MSC Handover Procedure*: This procedure takes place when the mobile terminal moves from a cell in MSC-A area to a cell in MSC-B area. MSC-A sends a *Perform_Handover* message to MSC-B [94], this message contains the parameters needed by MSC-B for allocating a radio channel and identifying the BS to which the call is to be handed over.

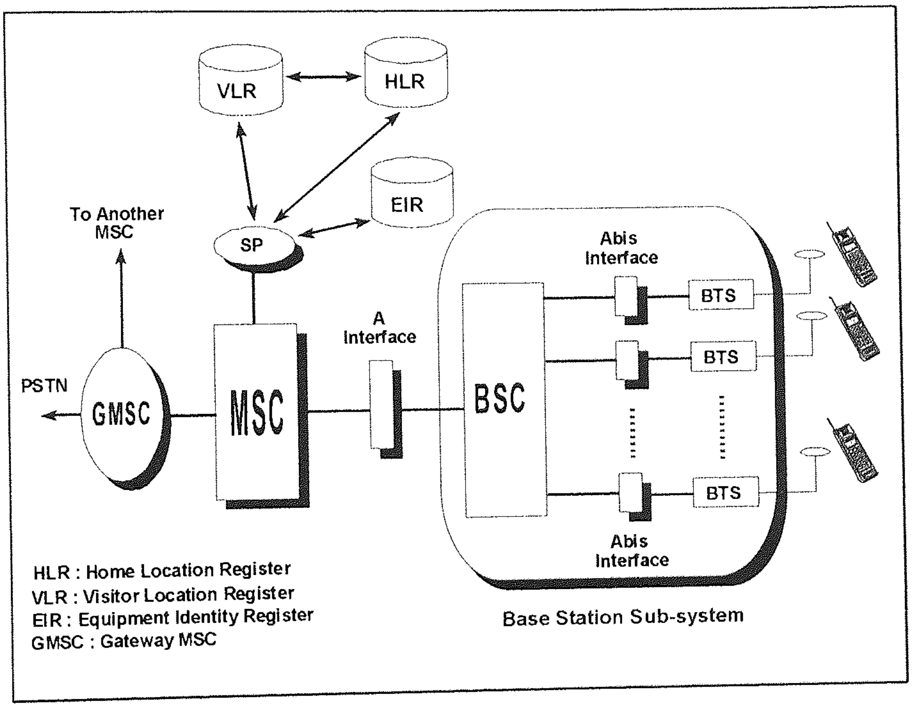


Figure. (5.1): GSM Public Land Mobile Network Configuration.

5.2.2 Handover Required Indication

The mobile terminal performs continuous measurements on the quality and the power level of the serving cell, and of the power level of adjacent cells. The measurement results are sent back periodically to the serving base station. When the network detects a deterioration in the perceived signal level from the mobile and realizes a need for handover, the BSS requests that a handover be carried out for that particular terminal. This is done by generating a *Handover_Required* message from the BSS to the MSC (see Figure.(5.2)). The *Handover_Required* message contains the following information: Message type, cause for handover, response request, preferred list of target cells, and radio environment information. If a handover is not possible, the MSC informs the BSS that the handover cannot be carried out. This is done by a *Handover_Required_Reject* message.

The *Handover_Required* message relates to a particular dedicated resource on the radio interface, either Dedicated Control Channel (DCCH[1]) or a Traffic Channel (TCH), and is sent using the Base Station Sub-system Application Part (BSSAP) connection already set up for that transaction. The BSS therefore continually monitors all radio information, and compares it with parameters such that if the transmission quality of a given parameter (or set of parameters) falls below a pre-determined threshold then a *Handover_Required* message is generated towards the MSC.

5.2.3 Handover Execution

Handover execution in the context of the BSS/MSC interface is the process whereby a MSC instructs a mobile terminal to tune to a new dedicated radio resource which may be on a different cell. The MSC sends a *Handover_Request* message to the target BSS from which it requires a radio resource (Figure.(5.2)). This message contains an indication of the type of channel required, and the terrestrial resource that will be used if the request is for a traffic channel. On receipt of this message, the BSS will choose a suitable idle radio resource. If queuing is managed, new requests which cannot be served immediately are put in the queuing file according to the indicated priority levels. If radio resource becomes available before the expiry of the queuing timer, this will be reflected back to the MSC in a *Handover_Request_Acknowledgement* message. The target BSS will then take all necessary action to allow the mobile terminal to access the radio resource that it has chosen.

The MSC receives the *Handover_Request_Acknowledgement* containing the appropriate channel, and the radio interface *Handover_Command* message is sent to the mobile terminal via the serving BSS. At the same time a timer T8 is started at the serving BSS. If at the target BSS a *Handover_Complete* message is not received before

[1] DCCH : Dedicated control channels are used for transferring registration procedure and call setup messages between the network and the mobile station.

the expiry of T8 (650 milliseconds) then the procedure shall be terminated, and the radio resource released for other calls i.e. by sending a *Clear_Request* message to the MSC. This allows for the mobile terminal to return to the old dedicated resource. The terrestrial resource will remain assigned until a *Clear_Command* is received from the MSC [95].

When the mobile terminal accesses the radio resource the target BSS checks the handover reference number to ensure that it is the same as expected, and hence there is a high probability that the correct mobile terminal has been captured (if the handover reference is not as expected then the BSS will wait for an access by the correct terminal). If the handover reference number is as expected, the target BSS will send a *Handover_Detect* message. When the mobile terminal is successfully in communication with the network, i.e. the *Handover_Complete* message has been received from the mobile, then the new BSS will immediately send a *Handover_Complete* message to the MSC. This terminates the procedure at the BSS. The dedicated radio resource and connected terrestrial resource will remain assigned until either the MSC instructs the BSS to release the resource or a reset occurs. If either a *Clear_Command* is received from the MSC, or a Reset is received from the MSC before a mobile terminal with the correct handover reference accesses the BSS then a handover resource allocation failure had occurred. A complete overview of the handover procedure is shown in Figure.(5.2) below.

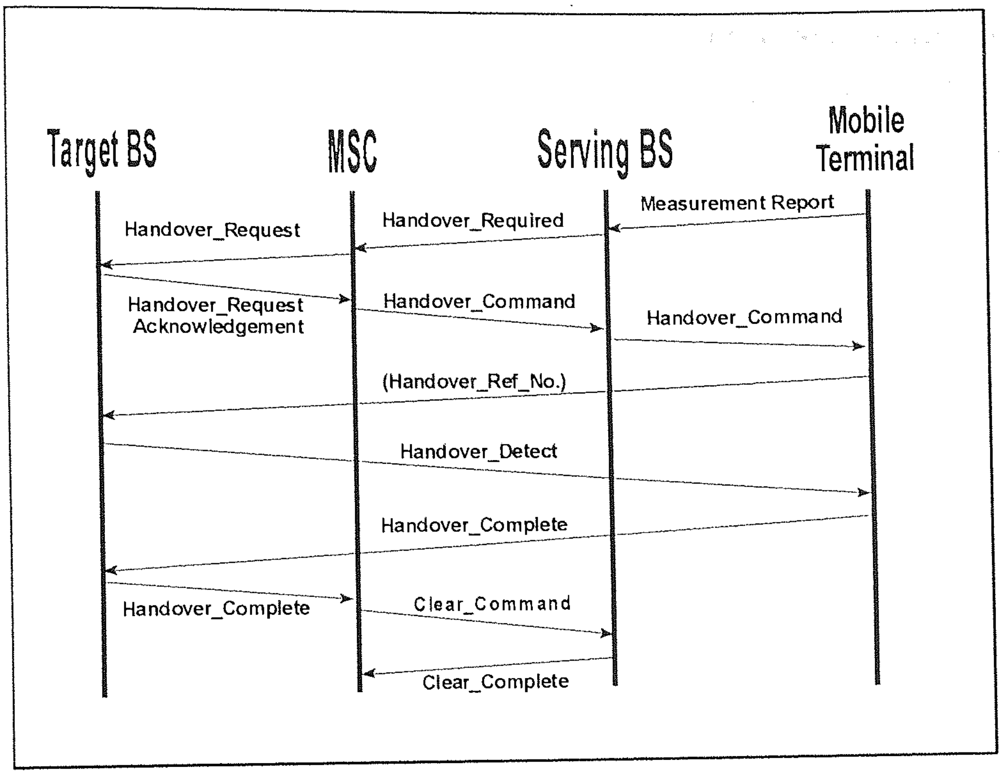


Figure.(5.2): Overview of the handover procedure between two Base Station Sub-systems on the same MSC.

If a *Handover_Failure* message with the cause value "radio interface failure, reversion to old channel" is received from the mobile terminal on the old channel then, a handover failure message is sent to the MSC. The generation of the *Handover_Failure* message terminates the procedure and allows all references to be released. The procedure at the target BSS is terminated by the MSC using a clear sequence.

5.3 Advanced Signalling Strategy

The objective of the new strategy is to reduce the signalling load handled by the fixed part of the PLMN during the handover process. This is achieved by re-routing the exchanged handover signalling, from the fixed part of the PLMN to the air interface via the mobile terminal. Such a substantial diversion would require modifications to the handover sequences already adopted in the GSM (explained earlier in Section 5.2.2) due to the change in the tasks and functional requirements of each element in the network.

The new signalling strategy introduces major changes to the responsibilities and the functions assigned to different entities in the PLMN. It aims to decentralize and transfer part of the control for the handover procedure from the fixed part of the PLMN to the mobile terminal. Effectively, the mobile terminal is assigned with an operational role greater than that in the current GSM standard. Unlike the current setup, setup where the handover procedure is initiated and conducted primarily by the network, in the new strategy the mobile terminal is expected to handle, process and initiate the handover procedures in response to the changing environmental conditions.

5.3.1 Improved Handover Procedure

To implement the new signalling strategy, it is required to change and re-allocate some of the functions performed by the fixed part of the PLMN to ensure correct handover operation. Figure.(5.3-a) shows the functions and the internal procedures relating to the handover process as adopted by the GSM standard, while Figure. (5.3-b) demonstrates the alterations introduced and the resulting set of functions as required by the new strategy. The following paragraphs will outline the main reason why these major changes were introduced.

In the new handover strategy the mobile terminal will be responsible for initiating the handover procedure. It will be exchanging handover signalling directly with the candidate cell by accessing one of the channels on that particular cell. The multiple access technique that would be used to allow the mobile terminal to perform such a task is the PRMA++ technique. This technique was chosen as the medium in which to implement the new strategy because it offers superior spectral capacity over conventional TDMA structure. A time slot in PRMA++ is reserved during activity periods only, while in TDMA the slot is reserved for the total period of the call. In effect this will give the mobile terminal a higher probability of success in attempting to reserve a slot on a channel in the new cell. Details of the frame structure and the access procedure will be explained later in Section 5.5.4.

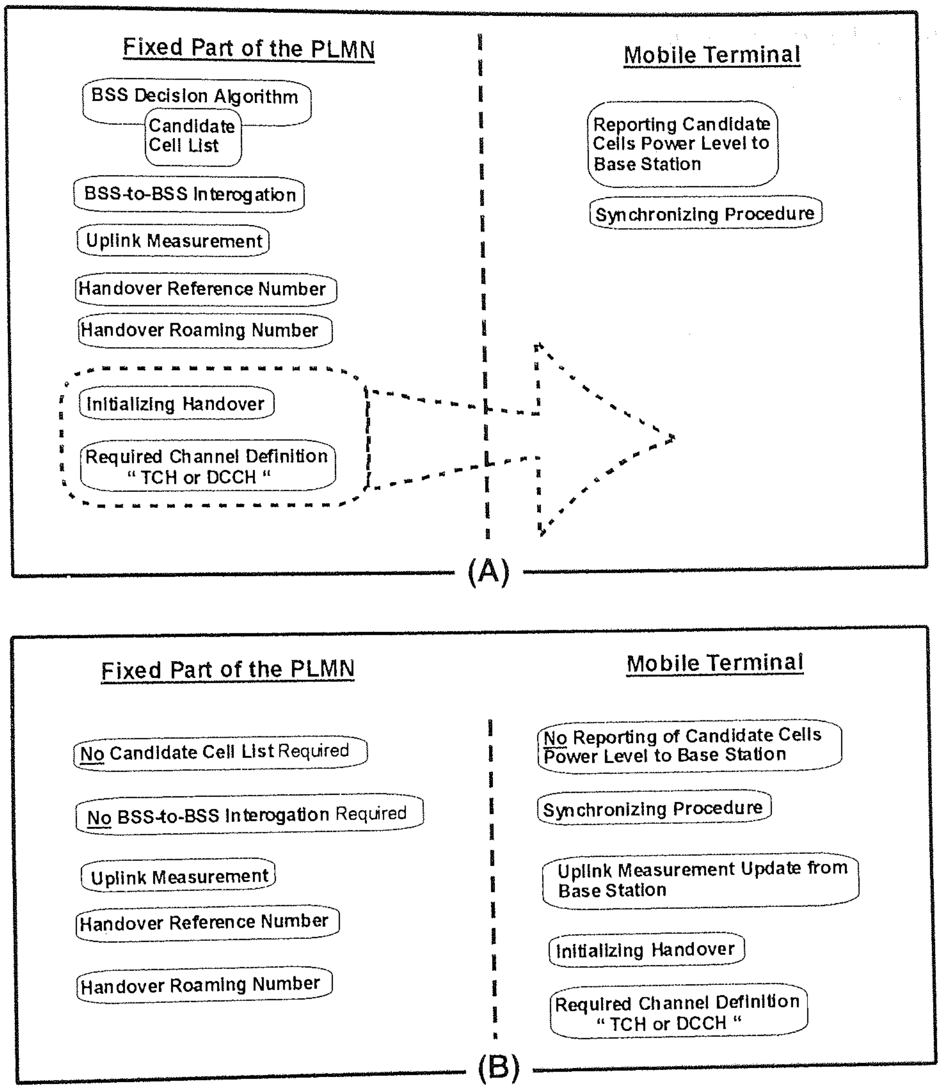


Figure. (5.3): The fixed network and the mobile terminal internal procedures required to perform inter-BSC handover in (a) GSM standard and (b) improved signalling.

As the mobile terminal is in communication with its serving base station, it will be carrying out measurements on the downlink transmissions from adjacent base stations. Instead of reporting these results back to the serving base station (as in the case in GSM), the mobile terminal is capable of compiling a list with a number of the up-to-date strong base stations that it had detected. The number on the list can be set according to the operation and maintenance (O&M) requirements.

The mobile terminal will be carrying out measurements on the downlink transmission from its serving base station. This is necessary to assist the terminal in initiating a handover on DCCH channel when the terminal is in an idle condition (i.e. no call in progress). Because the mobile terminal itself cannot determine the power level of its uplink transmission, the serving base station will continuously measure the power level of the uplink transmission (a process which already takes place in the GSM standard) and regularly send bursts on the downlink to inform the mobile terminal of its current uplink quality. If the quality or the power level drops below a certain threshold for uplink or downlink transmissions, the mobile terminal will initiate the handover procedure by attempting to access the first cell on the up-to-date list of candidate cells.

When the mobile terminal accesses a new cell, it will contend for a time slot in a similar fashion to the rest of the terminals in that cell. If successful, the mobile terminal will send a *Handover_Required* message indicating that the following procedure is a handover procedure. Consequently, the base station will reserve that particular slot for the rest of the handover transmission and will not release it for contention until a *Handover_Clear* command is sent to the MSC. The *Handover_Required* message will also contain an indication of the type of channel required and the terrestrial resource it will be connected to, for example whether it is a DCCH or TCH, and the number of slots required. If more than one slot is required as in the case of data transmission, the adjacent slots will be freed by the base station and allocated to the handover terminal. Freed slots will either be unoccupied slots or slots used by voice transmission only. Any packets already sent on them will be dropped and the relative terminals will have to contend for a new slot.

Upon receiving the *Handover_Required* message, the base station will send a *Handover_Request_Acknowledgment* message to the mobile terminal including a Roaming number and a *Handover_Reference* number. The Roaming number is required to re-route the call at the MSC to the new BSS. The Reference number is generated by the Visitor Location Register (VLR) of the new BSS, and it is used to reconnect the call to the mobile terminal at the end of the handover procedure. These numbers will be interrogated by the new base station to determine that they belong to this cell. This is important because the base station might have more than one handover procedure in progress at the same time. When the mobile terminal receives the *Handover_Request_Acknowledgment* message, it will generate a *Handover_Detail* message that includes the Base Station Identification Code (BSIC) of the last cell the terminal came from. This will assist the MSC in recognizing where the call is being transferred from.

Following the *Handover_Detail* message, the new BSS will issue a *Handover_Complete* command to the MSC upon which the call would be re-routed to the new BSS. A *Clear_Command* will be sent from the MSC to the old BSS to clear the call and release the channels allocated to it. Figure.(5.4) gives an overview of the new handover signalling procedure using PRMA++ .

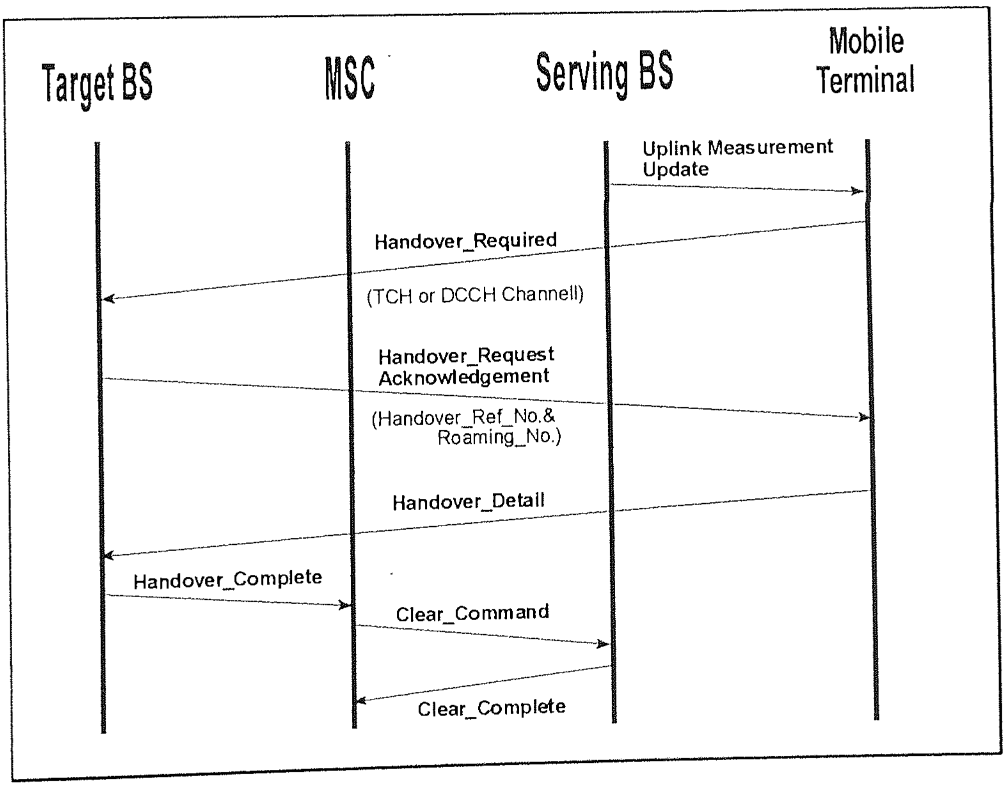


Figure.(5.4): Overview of the improved handover procedure using PRMA++.

Table (5.1) indicates the contents of the main three messages in the new handover procedure. The *Handover_Required* command issued by the mobile contains the type of channel required, i.e. whether it is for data transmission, voice transmission or an idle channel (when the terminal is not engaged in a call). However, in the GSM standard the fixed network (the serving BSS) is responsible for identifying the type of channel required for handover. The *Handover_Request_Acknowledgement* message contains the *Handover_Reference* number and the *Handover_Roaming* number which are issued by the new base station rather than by the MSC as in the GSM standard. This is because the control of the handover procedure is transferred from the MSC to the mobile terminal and the new base station. Also the mobile terminal is choosing its own candidate cell rather then relying on the serving BSS to compile a preferred cell list as it is the case in the GSM standard. The *Handover_Required_Acknowledgement* message also contains the BCCH of the new cell to enable the mobile terminal to transmit its control information to the new base station. It is important to mention that the BSIC of each cell should be included in the downlink transmission of that cell, otherwise the mobile terminal will not be able to distinguish whether the channel it is accessing is in a new base station or a channel in the old base station. The *Handover_Detail* message contains the BSIC for the old base station. This message is sent from the mobile terminal to the new base station to indicate the origin cell of the mobile. The old cell BSIC combined with the Handover_Reference number and the Handover_Roaming number will enable the MSC to re-route the call to the new base station and disconnect it from the old base station.

Message Name	*Contents*
Handover_Required	Type of channel required for handover, number of slots required.
Handover_Request_Acknowledgment	Handover_Reference_No. Handover_Roaming_No. BCCH of the Target Cell.
Handover_Detail	Originating Cell (i.e. serving cell) BSIC.

Table(5.1): Message contents for the new handover signalling procedure.

The new handover procedure bypasses the old BSS and therefore, eliminates the need for the old BSS to interrogate the new BSS for the availability of free channels. Utilizing PRMA++ as an air access technique means that there is no real need for a compiled list of candidate cells for handover, because the mobile terminal is accessing the channels in the new cell on a contention basis. Hence the BSS Decision Algorithm

[96], where the BSS should be capable of compiling a list of candidate cells, will not be required. The mobile terminal will only be denied access to all the channels in the new cell if all the slots were already occupied with active terminals transmitting packets. This situation can only occur for a short time when voice terminals are involved due to the inherent fluctuating nature of speech patterns.It is more likely to occur if a majority of the traffic being handled by the new cell is data traffic. Even with intracell and intercell handovers within the same BSS, the mobile terminal can access the available channels without the need to contact the base station, or this procedure can be initiated by both the mobile terminal and the base station, allowing priority to the mobile terminal.

5.4 COMNET III Simulation

The network simulated in COMNET III is used to examine the handover signalling and its effect on the transmission of speech packets in a heavily loaded GSM cellular network. COMNET III offers the capability of monitoring single packets from the instant of generation at the origin point to the time of delivery at the destination point. This allows the determination of the time delays endured by individual speech and handover packets during the handover procedure. Figure.(5.5) shows the network simulated with COMNET III. It is a typical GSM PLMN comprising physical units, interfaces and transmission rate parameters. However, it differs from the standard GSM PLMN configuration in that this model is a multi-tier network supporting both macrocells and microcells.

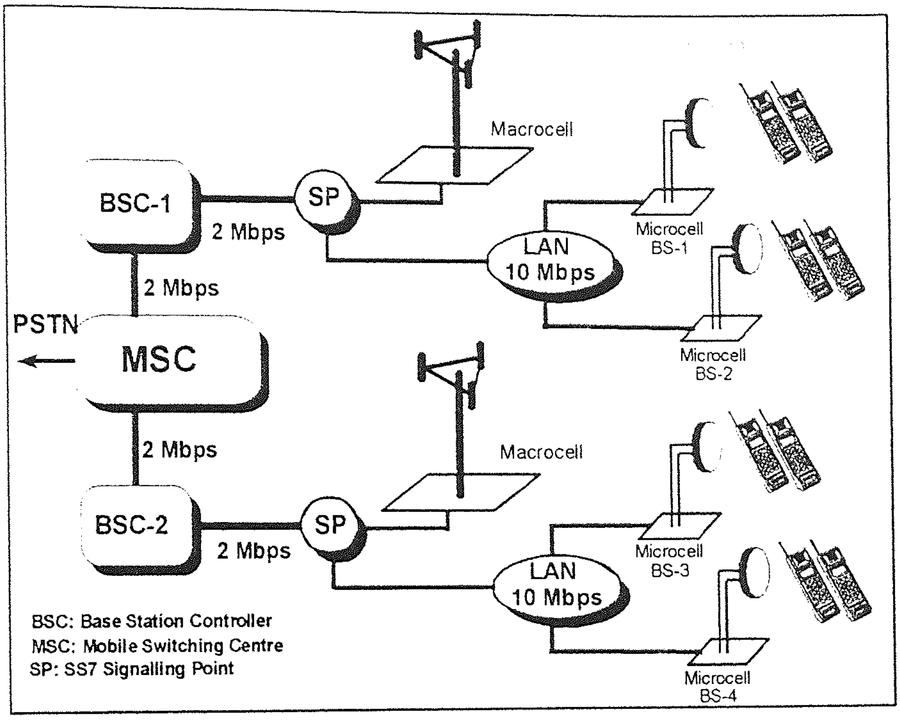


Figure.(5.5): A multi-tier cellular network simulated using COMNET III.

Although the COMNET III software is capable of specifying fixed network parameters and transmission protocol specifications with considerable detail, the software is only capable of simulating point-to-point air link transmission in terms of bit rates, with no facility available for medium access technique representation. Therefore, the simulated network model was assigned all the parameters relating to the fixed part of the GSM-PLMN, but it was only possible to represent the TDMA medium access operation in terms of regular packet generation within a specific time frame. This was performed assuming that mobile terminals were close enough to the base stations and the air link did not suffer from shadowing or multiple fading errors.

The network model consisted of four microcells representing the lower layer in the multi-tier network. Each pair of microcells was connected to a LAN. The LANs in turn were connected to macrocellular base stations representing the second layer in the cellular structure. Each microcellular base station was allocated a single channel serving eight mobile terminals, which is the maximum number of users that can be accommodated on a GSM air frame. Since COMNET III offers a versatile range of parameters for designing transmission links in fixed networks, it was possible to simulate a heavily loaded fixed network by applying variable loading levels to the links without physically creating every single terminal in the simulated coverage area. All the calls considered in this model were mobile to PSTN calls; mobile to mobile calls were not included.

As shown earlier in Figure.(5.2), the inter-BSC handover procedure is dependent on two major parts. First is the availability of a free channel in the target BSS and the time spent to allocate that free channel to the mobile. This is determined by the time elapsing from the instant of generating the *Handover_Required* message at the serving BSS to the instant where the *Handover_Command* message is received at the mobile terminal. Second, is how fast the mobile terminal physically accesses the free channel and performs the handover; this is determined by the time period between receiving the *Handover_Command* at the mobile terminal, and issuing the *Clear_Command* message at the MSC. Figure.(5.6) shows the first part of the handover signalling which the COMNET simulation deals with. At this stage the availability of a free channel in the target base station is indicated in the *Handover_Required_Acknowledgement*. The time period required to obtain this acknowledgement is greatly affected by the level of signalling imposed on the fixed network where heavy teletraffic conditions result in higher delay figures. It is vital to mention this because the handover process is initiated according to the most recent power and quality levels of the uplink transmission. It was decided that in this simulation the handover procedure would be triggered with the last transmission received from the mobile terminal.

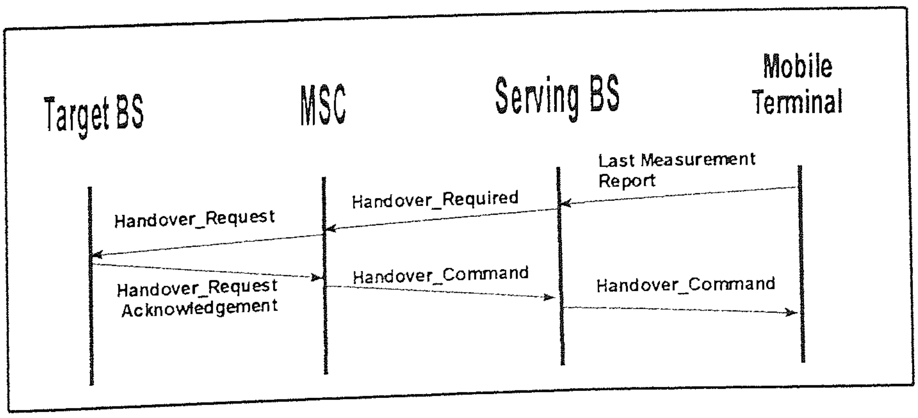


Figure.(5.6): Free channel assignment at the target BSS in GSM.

5.4.1 Design Parameters

In this simulation, mobile terminals were assumed to be always in communication with the microcellular base station. During a handover, the traffic channel (TCH) is substituted by the Fast Associated Control Channel (FACCH) [97,98]. Therefore for each terminal, speech packets and handover packets use the same channel and time slot. To simulate a TDMA time frame, each terminal was represented by a speech packet source. Packets are generated at an inter-arrival time equal to the period of the TDMA time frame, which is 4.62 milliseconds (8 slots). The packet size was chosen equal to the number of bits comprising the time slot, i.e. 156 bits as shown in Figure.(5.7). Although only the coded bits are used for carrying signalling and speech information, the rest of the slot had to be accounted for since it includes the channel coding for the TDMA transmission. In the fixed network, the same packet size was used for handover signalling to include transmission routing overhead.

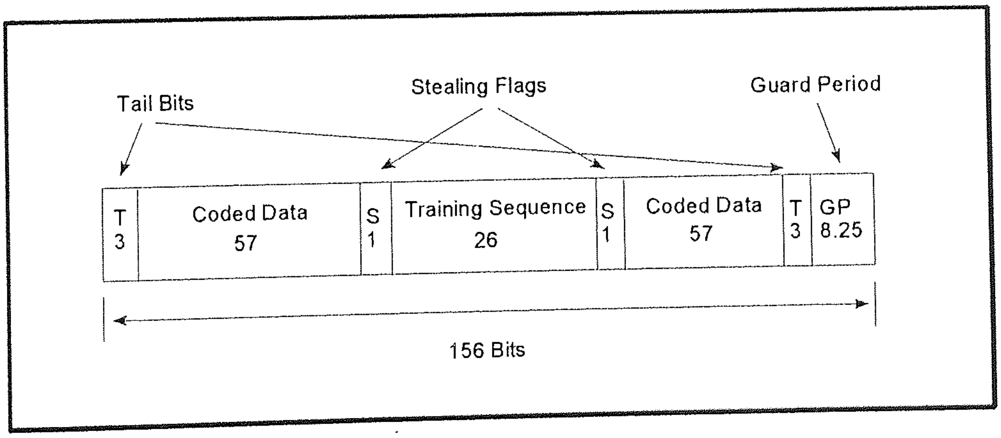


Figure.(5.7): Structure of a normal burst in TDMA slot.

Handover requests were generated at a rate proportional to the proposed terminal speed when passing through the microcell. A mobile terminal handing over from one microcell to another will vacate a time slot on the channel of the first microcell. Since a maximum loading situation is assumed, a vacant time slot will be filled again by an active call that has either been initiated from the first microcell, or was handed over from another microcell. The handover signalling is generated assuming that mobile terminals are continuously passing through the microcell and handing over to the adjacent base stations. Handover packets were assigned higher priority than the speech packets, i.e. when queuing in a certain node's buffer; the handover packet will always be advanced forward ahead of the rest. If a speech packet was already being processed, the handover packet cannot abort the operation and it would have to wait until the latter operation is finished.

5.4.2 Results and Performance Analysis

For the network shown in Figure.(5.5) four mobile terminals were handed over from microcell BS-1 to microcell BS-3. The handover signalling was generated at an exponential rate with a mean value of 6 seconds. This value was chosen to represent a multi-terminal mobile platform crossing a microcell at an average speed of 20 km/h, in terms of metric units this becomes:

mobile station moving at 20 km/h ==> 20 000 m/ 3600 s ==> 5.5 m/s

Thus in 6 seconds; a mobile terminal will be crossing a distance of 5.5 x 6 = 33 m, which was chosen as the average diameter of a microcell in dense urban areas. The handover rate could be increased or decreased to represent faster or slower moving terminals respectively.

When BS-3 receives a handover packet transmitted from BS-1, a response packet will be generated and sent back to BS-1. The response packet will be sent across the same route over the fixed network. The simulation was executed for 90 seconds. This period was considered to be adequate to reflect a picture of the packet delay figures during the handover process.

Figure.(5.8) shows the obtained actual values and the cumulative mean values for the inter-BSC *Handover_Request* packet delay. This is the time delay required by the packet to traverse from the mobile terminal to BS-3, via BS-1, BSC-1, MSC and BSC-2. The minimum value recorded for the handover delay is 0.913 millisecond, which is the minimum transmission delay a packet will endure if no further delays were included at the network nodes. This can be calculated as follows:

$$\text{Transmission delay (in seconds)} = \frac{\text{Packet size (bits)}}{\text{Transmission rate (bit per second)}} \qquad (5.1)$$

Hence,

$$\text{Packet delay} = \frac{156\ (\text{bits})}{270\ (\text{kbps})} + 2\times\left[\frac{156\ (\text{bits})}{10\ (\text{Mbps})}\right] + 4\times\left[\frac{156\ (\text{bits})}{2048\ (\text{kbps})}\right]$$

$$= 0.913 \times 10^{-3} \text{ seconds}$$

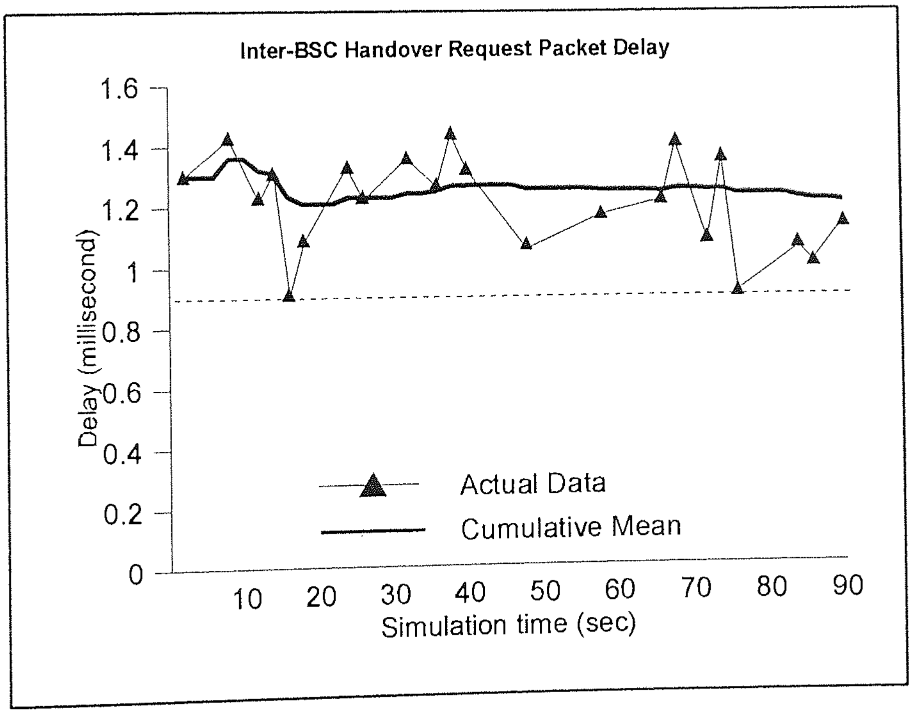


Figure.(5.8): Inter-BSC Handover_Request packet delay.

Due to the heavy teletraffic loading in the fixed part of the PLMN, the maximum values for handover delay changes greatly due to the queuing in the network nodes. Additional delay occurs due to the fact that even though a handover packet has a higher priority in queuing, it cannot pre-empt a speech packet in progress. Therefore the

Handover_Request packet will have to wait until the speech packet processing is complete.

The effect of the resulting delays can be determined by looking at the RF activity between the mobile terminal and the base station in Figure. (5.9). The mobile terminal transmits a packet on the uplink slot and awaits for a period of four slots, after which it will receive the transmission from the base station on the down link. An optimum signalling situation is when the mobile transmits a *Handover_Request* packet on its uplink slot, and receives the desired *Handover_Response* packet four slots later on its downlink slot. Taking in consideration the fact that handover signalling cannot be cancelled or predicted, as is the case with the speech packets, any delay in delivering the *Handover_Response* packet will result in the mobile terminal waiting for a the next time frame to receive the desired response and consequently jeopardising the success of the handover procedure.

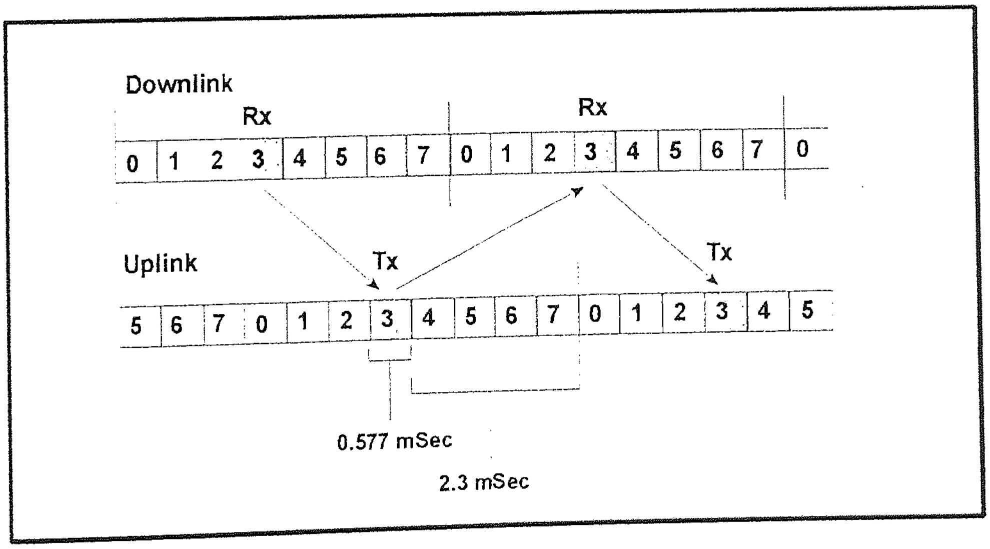


Figure.(5.9): RF Activity between the base station and the mobile terminal.

From Figure.(5.8) the minimum delay of the *Handover_Request* packet is 0.913 ms, which if the same delay was encountered on the route back for the *Handover_Response* packet, then the round trip delay would be 2 x 0.913ms = 1.826 ms which is within the four time slot limit (2.3 ms).

However, it is clear from Figure.(5.8) that the maximum values of the *Handover_Request* packet delay are many times above 1.15 ms (1.15 ms would give a round trip delay of 2.3 ms). It is also the case with the cumulative average, which is slightly above the 1.2 ms value, giving a round trip delay of 2.4 ms minimum, which is again outside the four slot limit.

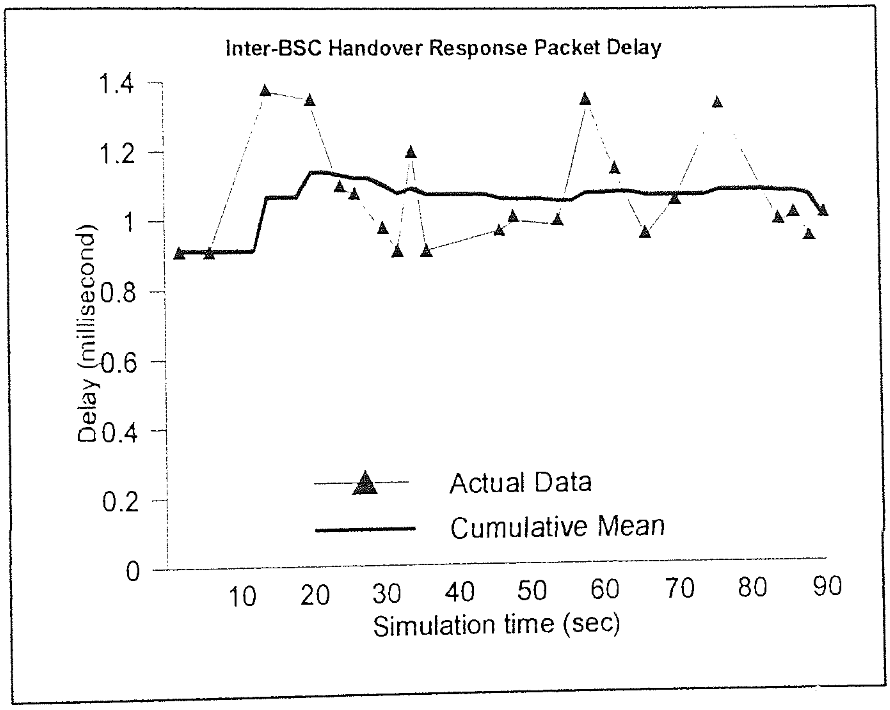


Figure.(5.10): Inter-BSC Handover_Response packet delay.

Figure.(5.10) shows the delay for the *Handover_Response* packets generated at BS-3, this is the equivalent of the *Handover_Request_Acknowledgment* and the *Handover_Command* messages in Figure.(5.6). The minimum value of 0.913 ms is recorded. Maximum values of up to 1.4 ms were also recorded which, if the same delay were to be encountered in the *Handover_Request* packet, would lead to a round trip delay beyond the four slot limit. However, it is noticeable that on average the *Handover_Response* packets have less time delay than that encountered in the inter-BSC *Handover_Request* packet. A mean value of around 1.1 ms was recorded during the majority of the simulation time.

This is due to the fact that when a *Handover_Request* packet is generated and delivered to BS-3, the rest of the TCH transmission with BS-1 (i.e. the time slot on the uplink and the downlink) will be used by the FACCH for the handover signalling. The slot on the downlink will thus be already reserved for the *Handover_Response* packet and no contention with speech packets will be required, which is the case with the *Handover_Request* packet.

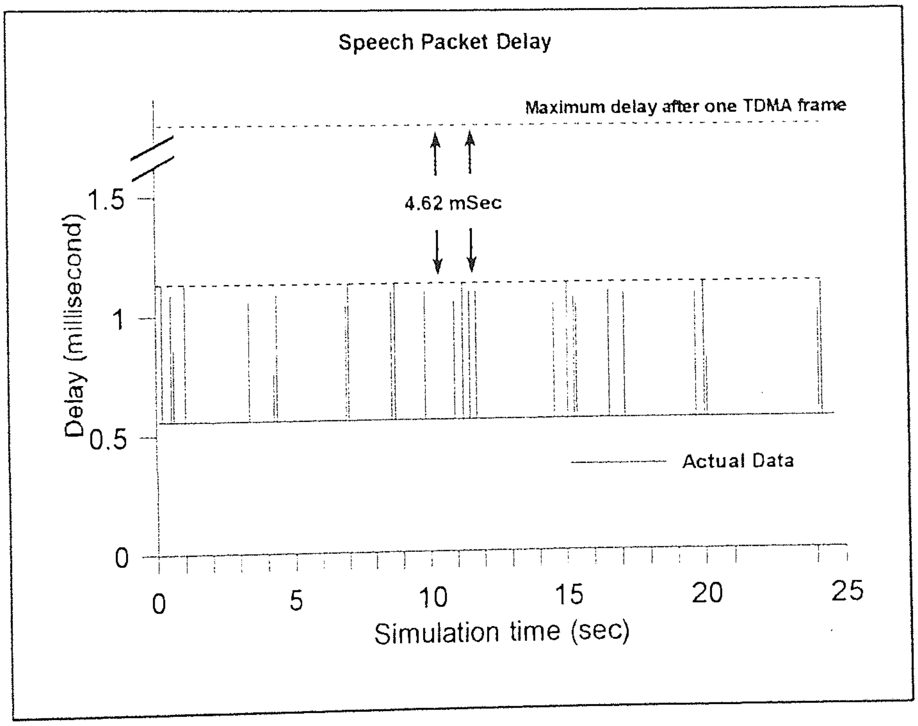


Figure.(5.11): Speech packet delay on the uplink.

Figure.(5.11) shows the time delays encountered by the speech packets for a simulation period of 25 seconds rather than 90 seconds as shown previously. This is because the time difference between the consecutive packets is far too small to be indicated individually on the plotting graph (TDMA time frame period of 4.62 ms), and therefore the packets are represented by a solid line. The horizontal line at the value of 0.577 ms represents a transmission delay of a 156 bit packet over the 270 kbps air link, which is the delay for the majority of the speech packets. The vertical lines represent speech packets that suffered delay greater than the usual air link transmission delay due to the coincidence with higher priority *Handover_Request* packets. In Figure.(5.11) the maximum delay value a speech packet had encountered is equal to the air link transmission delay (i.e. of 0.577ms) plus an extra 0.577 ms, which is the time delay required to transmit a *Handover_Request* packet. Such delays occur when a speech packet is generated at an instant where the time slot is already occupied with the *Handover_Request* packet. The speech packet cannot be transmitted until the *Handover_Request* packet is processed and transmitted.

In a typical TDMA transmission, when the speech packet is delayed by the *Handover_Request* packet, the speech packet will either be discarded or buffered for re-transmission on the same slot in the next TDMA frame. Therefore, any speech packets delayed beyond the air link transmission delay of 0.577 ms would have a delay figure of 4.62 ms added to it, which is the period of a complete TDMA time frame (see Figure.(5.9)). However, in this simulation a delayed speech packet is transmitted as soon as the *Handover_Request* packet is processed and transmitted, therefore the delay figures obtained here do not include the buffering time of waiting for the next TDMA frame. In Figure.(5.11), the upper dotted line indicates the maximum delay value if buffering delay were to be included.

The reason behind ignoring the buffering delay and choosing the form of representation used in Figure.(5.11) was because speech packets will only be delayed in the first stage of the handover process (i.e. when the *Handover_Request* packet is sent). Hence the results indicated in Figure.(5.11) are for the speech packets of an active terminal in microcell BS-1 up to the first stage of the handover process. After that it would be the delay for speech packets of another mobile terminal that occupied the vacated time slot. Also if speech packets were to be discarded as soon as they suffered any delay, it would not be possible to determine the pattern of delays imposed by the handover signalling on the normal transmission of the speech packets.

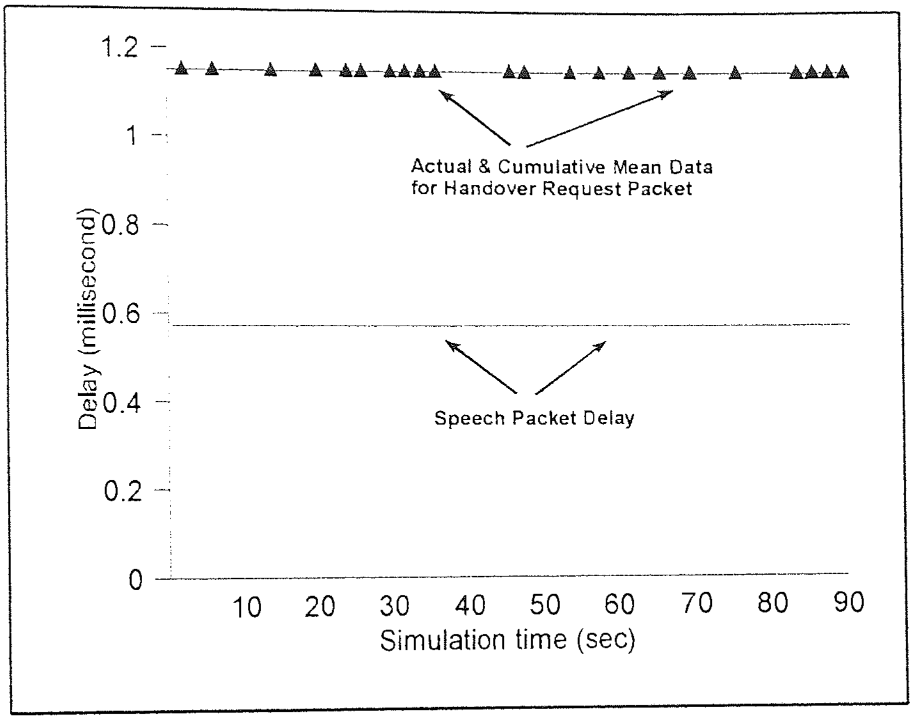


Figure.(5.12): Inter-BSC Handover_Request & Speech packet delay with air interface signalling.

For the mobile terminals handing over from BS-1, an air link was established between these terminals and the target base station BS-3. Handover signalling between the old and the new base stations was diverted from the fixed network to the air interface via the mobile terminal. The result shown in Figure.(5.12) demonstrate the time delay for the *Handover_Request* packet generated from BS-1 and sent to BS-3 via the mobile terminal. The packets have a constant delay value of 1.154 ms which is twice the air link transmission delay of a 156 bit packet (156 b/270 kbps = 0.577 ms). This value was the same for the *Handover_Response* packets which were transmitted from BS-3 to BS-1 via the mobile terminal. Also shown in Figure.(5.12) is the uplink transmission delay for the speech packets from the mobile terminals to BS-1. The round trip delay for sending a Handover_Request packet and receiving a *Handover_Response* packet is twice that indicated in Figure.(5.12), i.e. 2x1.154 ms = 2.3 ms. Therefore provided that the response obtained from BS-3 is immediate then the *Handover_Response* packet will be received within 4 time slots from the instance of sending the *Handover-Request* packet.

It is clear from Figure.(5.12) that diverting the handover signalling to the air interface not only reduces the total delay required for achieving handover, but it also results in a constant level of delay for the handover packets as well as the speech packets. This is because the *Handover_Request* packet and the *Handover_Response* packets were sent over channels separate from that used by the speech packets. Therefore there was no contention between the packets of different types. Also, the handover packets would not have to pass through the fixed network and therefore they would not be affected by the network congestion and the level of signalling handled by the PLMN.

Table (5.2) below presents a summary of the results obtained in the COMNET III simulation where the total delay required for sending the *Handover_Request* packet and receiving the *Handover_Response* packet is indicated for two cases. The "Best Case Delay" considers the total delay as double the maximum value recorded for the cumulative mean of the *Handover_Request* packet delay in Figure.(5.8). The "Worst Case Delay" considers the total delay as double the maximum value recorded for the actual data in Figure.(5.8). Table (5.2) demonstrates that for handover signalling via the fixed network, the total delay in both Best and Worst cases exceeded the 4 time-slot limit (2.3 ms). Therefore the current signalling route would result in delayed responses and hence an overall delay for the handover procedure. Even if the Best case was to consider the average value of the cumulative mean instead of its maximum value (around 1.25 ms from Figure.(5.8)), then this would result in a round trip delay of 2.5 ms which is beyond the 4 time slot limit. For signalling via the air interface, the Best and the Worst case transmission delays are constant and exactly equal to 4 time-slots. This enables the receipt of the *Handover_Response* packets within 4 time slots from sending the *Handover_Request* packets.

The model simulated in COMNET III did not include any extra delays for packet processing and handling at the different nodes in the network. The delay figures obtained from this simulation were mainly queuing delays due to the high levels of teletraffic handled by the network. Any delays relating to packet processing operations and procedures will add to the figures already obtained. In this case the handover signalling via the fixed network will be mostly affected due to the number of nodes that handover signalling packets have to traverse on route to their destinations. For the handover signalling via the air interface, the only delay that could be added is the time required by the target BSS to provide the necessary information to be conveyed to the mobile terminal in the *Handover_Response* packet.

Handover Signalling	**Handover_Request Packet Delay (mSec)**		**Best Case Total Delay (mSec)**	**Worst Case Total Delay (mSec)**
	Max. Mean	**Max. Actual**		
via *Fixed Network*	1.36	1.45	2 x 1.36 = 2.72	2 x 1.45 = 2.9
via *Air Interface* (Mobile Terminal)	1.154	1.154	2 x 1.154 = 2.308	2 x 1.154 = 2.308

Table (5.2): Results summary indicates a significant reduction in the worst case delay.

Transferring the handover signalling to the air interface requires a certain mechanism to enable the dual communication between the mobile terminal and both the old base station BS-1 and the target base station BS-3. Such a mechanism would ensure that the mobile terminal does not communicate with both station in the same time slot. Also it would require the mobile terminal to acquire a level of functional responsibilities higher than that currently maintained in the GSM standard. The COMNET III simulation presented above concentrated on demonstrating the effects of implementing such a mechanism rather than explaining or simulating the mechanism itself. The following sections will deal with the simulation of the medium access techniques and will examine in full detail the specification and implementation of the new handover mechanism. Also the next sections will analyze the new handover mechanism with regard to its effects on the capacity of the system and the quality of the overall performance.

5.5 Simulation of Medium Access Protocols

In order to investigate the performance of different medium access techniques and introduce improved and enhanced schemes it is vital to have a detailed model of the network under consideration. This part of the simulation was performed to measure the time delays and evaluate system performance associated with the complete handover procedure in the microcellular network. TDMA and PRMA medium access protocols have been utilized to compare performance of both protocols when applying the standard and the improved handover signalling strategy. Unlike the COMNET III model, where the mobile terminals were presented as a percentage of teletraffic loading applied to the fixed network links, the microcellular network demonstrated in this part considered every mobile terminal as an individual entity. Hence it was possible to evaluate the effect of every terminal on the overall performance.

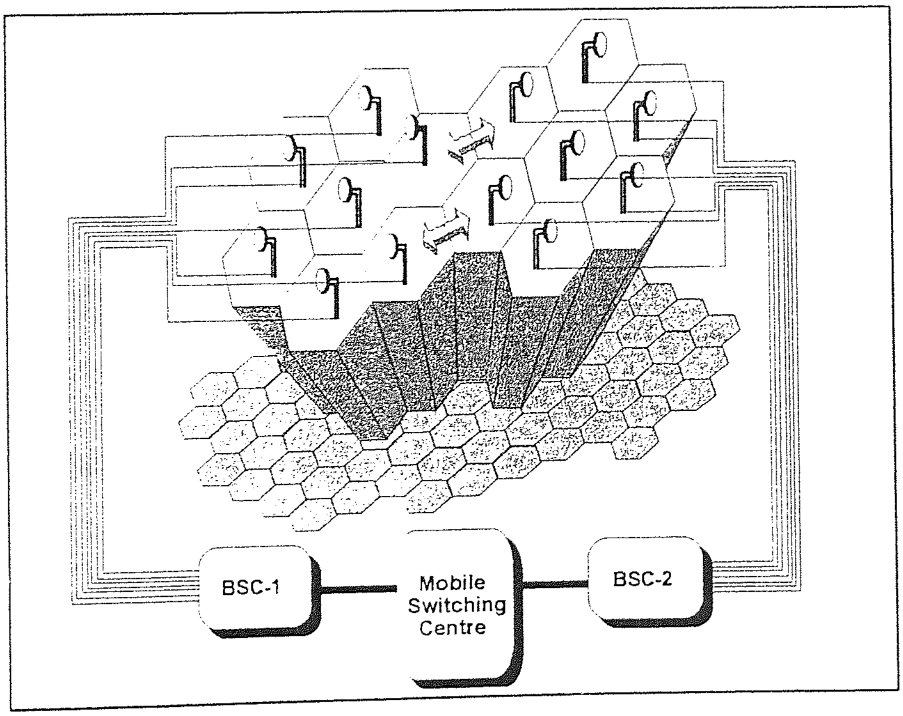


Figure.(5.13): Microcellular network with 2 clusters controlled by two separate BSCs.

The network model shown in Figure.(5.13) consisted of two clusters controlled by two separate BSCs. As in the case with COMNET III, the handovers of interest are the inter-BSC handovers, indicated by the double arrows in Figure.(5.13) on the inter-BSC microcells[2]. Intra-BSC handovers (i.e. handovers between microcells controlled by the same BSC) are not taken into account when the delay measurements are performed. This is because such handovers are carried out by the controlling BSC and usually executed in a much shorter time period than inter-BSC handovers.

5.5.1 Network Configuration

Parameters for the modelled microcellular network are listed in Table (5.3).

Number of BSCs	2, each controlling a cluster.
Cluster number & size	2 clusters with 7 cells each.
Cell type	Hexagonal.
Average cell diameter	30, 40 & 50 meters.
Access technique	TDMA, PRMA++, Advanced PRMA
Channel Assignment	FCA, DCA
Call arrival	Poisson distribution.
Call duration	Negative Exponential distribution.
Calls direction	Mobile to PSTN only.
Mobile terminals distribution	Uniform across the coverage area.
Mobile Terminal speed	10-50 km/h, at steps of 10 km/h.
Handover type	Inter-BSC only.

Table (5.3): Network design parameters for access technique modelling.

The modelled network represents enclosed areas inside buildings, underground stations, underground shopping centres or even enclosed areas between tall buildings. This type of microcell shares the disadvantage of being shadowed from the macrocell antenna transmission and therefore creating gaps in the cellular coverage. Maintaining a call in this type of cellular layout might not be a problem for pedestrian-held mobile terminals, but the same cannot be said for vehicular based terminals passing through such an environment. Active mobiles roaming at speed through the microcellular area will

[2] Inter-BSC microcells refer to the two pairs of microcells from the two clusters where the handovers are taking place.

require efficient microcell-to-microcell handovers, something which will impose a considerable signalling load on the fixed part of the PLMN, particularly when numerous voice and/or data calls have to be transferred simultaneously.

Microcells vary in shape and size depending on the environment they are deployed to cover. For simulation purposes the microcells were assumed to be hexagonal with average cell diameters of 30, 40 and 50 meters. Calls considered in this simulation are initiated from mobile terminals to PSTN only, mobile to mobile calls are not included. This is due to the possibility that two terminals in communication could move to the same cell or two cells controlled by the same BSC and effectively reduce the loading on the fixed network links. Also, the termination of a mobile-to-mobile call even when the two terminals are in different BSC cells will remove two calls from the signalling load instead of one, effectively distancing the network from the maximum loading condition which is an important and critical feature for this simulation.

Mobile terminals are distributed uniformly across the network. Calls are generated according to a Poisson distribution and the call duration follows a negative Exponential distribution with a mean value of 90 seconds. The teletraffic load offered to each microcell ranged from 0.6 Erlang per terminal to 3 Erlang per terminal, which is the equivalent of 150 seconds to 30 seconds of call inter-arrival time per terminal; respectively. The above can be explained by considering the Erlang definition [99]:

$$E = \lambda h \qquad (5.2)$$

where: E = traffic in erlangs,

λ = mean call arrival rate (i.e. calls per unit time) = 1 / call inter-arrival time,

h = mean call holding time,

A call inter-arrival time of 60 seconds per terminal is equivalent to a call arrival rate of 1 call per minute per terminal, which is equivalent to 0.0166 call per second per terminal. Hence by applying equation (5.2) with a mean call duration time of 90 seconds:

Erlang per terminal = 0.0166 x 90 = 1.5

Since the simulation is mainly concerned with inter-BSC handovers, only terminals crossing borders between inter-BSC cells are considered for handover statistics; intra-BSC handovers are treated as new calls. Inter-BSC handover requests are generated following a Poisson arrival rate with a mean value depending on the cell size and the speed at which the terminal is moving. Speed ranged from 10 km/h to 50 km/h, increased in steps of 10 km/h. Terminals in the inter-BSC microcells were assumed to be moving at the same speed to represent active mobile terminals situated on moving platforms like trains or cars passing through the microcellular area. For this type of speed the handover performance is quite critical and relies on the availability of free channels, as well as on the time delay taken to transfer the calls to the new channels before the connection deteriorates and is lost by the old base station.

5.5.2 Parameters for TDMA Operation

Table (5.4) shows a list of the design parameters used for simulating TDMA operation. Carrier bit rate and air frame specifications were to typical GSM standard specifications. The coded data bits, training sequence bits and the tail bits constituting the contents of the time slot were all considered as one complete packet, as shown in Figure.(5.7). Packet size for the air interface transmission and the fixed network transmission was chosen as 156 bits.

Medium access	TDMA.
Channels per cell	16
Slots per airframe	8
Frame length	4.62 milliseconds.
Carrier rate	270 kb/s.
Packet size (bits)	156
Channel Allocation	FCA, DCA.
Voice activity detection	None

Table (5.4): Design parameters for TDMA operation.

Although in real networks the number of channels allocated per cell could differ from one cell to another (depending on the number of mobile terminals expected to be served in the different cells), in this simulation each cell was allocated 16 channels to conform with the GSM standard. Because the microcell size is small, cell sectorization was not implemented and channels were filled in ascending order, i.e. the first free channel on the 'channel-per-cell' list is allocated to the first new call request. To achieve a maximum loading scenario, each microcell was populated with 128 mobile terminals at the start of the simulation. This is the maximum number of the terminals that can be served with the 16 channel set.

The channel assignment schemes implemented over TDMA were FCA and DCA. In the FCA each channel can be equally assigned to the handover traffic entering the microcell or to new calls initiated within that microcell. In the DCA, each microcell was allocated 16 channels to start with. If the cell requires more channels, an extra channel is allocated dynamically only if that channel satisfies the co-channel interference requirements, i.e. the nearest microcell with a similar channel should be separated by a minimum distance D_S, shown in Figure.(5.14), which is defined as follows [100]:

$$D_S = R\sqrt{(3K)} = 4.6\,R \qquad (5.1)$$

where R is the cell radius and K is the cell re-use factor (K=7 in this case).

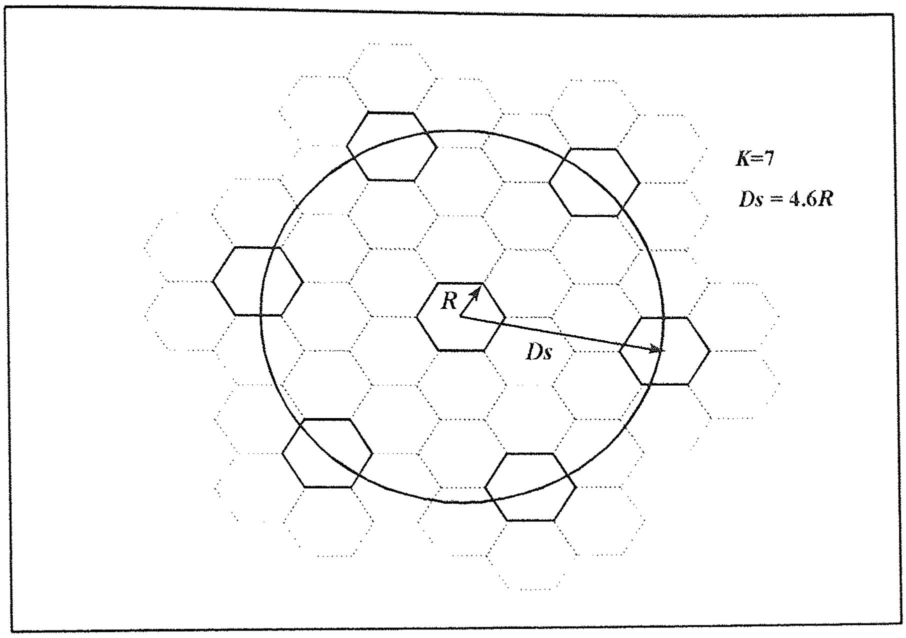


Figure. (5.14): The minimal separation (D_S) required between any two co-channel cells.

Dynamic channel assignment was implemented in three priorities: (1) channels can be assigned dynamically to handover requests as well as new call requests in an equal probability, (2) channels are assigned dynamically for handover requests only (this will be performed merely in the four inter-BSC cells where handovers are taking place), and (3) channels are assigned dynamically to new call requests only, while handover requests will only be allocated channels from the primary 16 channels set. The complete GSM handover procedure (mentioned earlier in Section 5.2.3) is adopted for inter-BSC handovers in the TDMA network.

5.5.3 Parameters for PRMA++ Operation

The network structure shown in Figure.(5.13) was simulated with PRMA++ medium access specifications [101]. The parameters implemented for the PRMA++ protocol are listed in Table (5.5). It is vital to mention that although the uplink frame in the PRMA++ standard comprises 72 time slots [102], in this simulation the uplink frame consisted of 46 time slots in order to reduce simulation time. The 46 slots contained 30

I slots (information slots) and 16 R slots (reservation slots), bringing the frame length to 3.22 milliseconds. An information slot contains 66 user data bits and 59 bits for burst overhead including guard bits, training sequence, and inband signalling as shown in Figure.(5.15) [101,102]. The carrier bit rate is 1.8 Mbps and each microcell was allocated two carriers. The downlink frame has a similar number of slots, but reservation slots are substituted by A slots (acknowledgement slots). When a mobile terminal makes a successful reservation on a certain R slot, the base station will transmit the mobile's address and the allocated uplink slot on the corresponding down link A slot.

Medium access	PRMA.
Channels per cell	2
Slots per frame	46, 30 Info & 16 Reservation.
Frame length	3.22 milliseconds.
Packet dropping threshold	2 frames.
Carrier rate	1.8 Mb/s.
Packet size (bits)	125
Voice activity detection	Yes

Table (5.5): Design parameters for PRMA++ & Advanced PRMA operation.

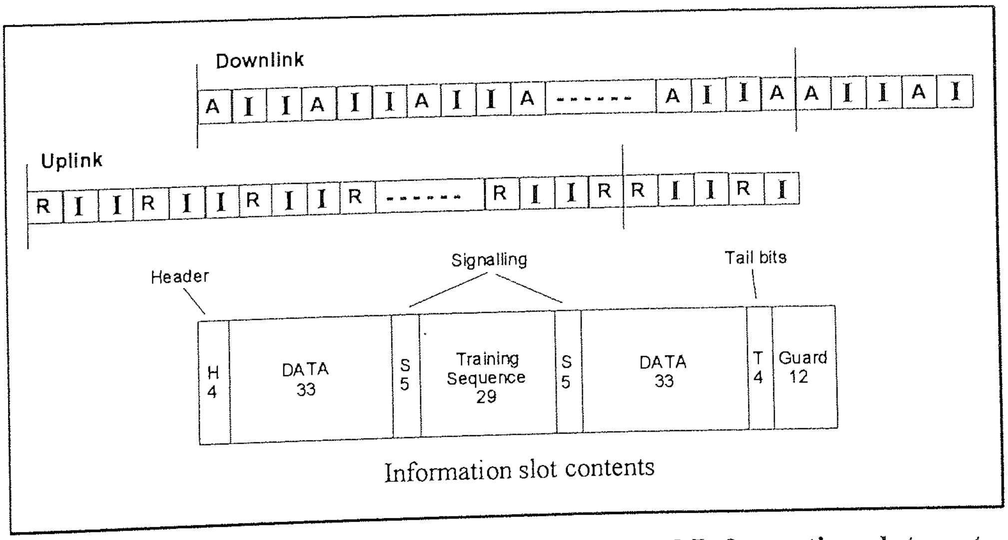


Figure. (5.15): Advanced PRMA frame structure and Information slot contents .

Mobile terminals reserve time slots on the basis of speech activity i.e. when there are speech packets to transmit. The packet will be queued for a period of two frames, after this period the packet will be discarded and the mobile will start contending with the next packet. Once successful, the mobile terminal will retain that uplink slot for the rest of the talkspurt. At the end of the talkspurt or speech activity period, the information slot is released by transmitting a corresponding release message to the base station.

As a minimum load, each cell was assumed to be populated with 60 terminals, which is the number of slots available on the two carriers and therefore terminals can operate without any contention. Terminal population is increased in steps and the system's overall performance is observed. For inter-BSC handover, the complete signalling sequence for the GSM standard is followed. However, in PRMA++ the mobile terminal will have to contend for a time slot to send its *Handover_Required* message; once a time slot is granted it will be reserved for the mobile terminal to resume its handover signalling. The fact that a slot is reserved for the period of the handover might reduce the contention delay expected every time the air interface is accessed, but the handover process has to withstand the signalling load across the PLMN.

5.5.4 Signalling Mechanism in the Advanced PRMA

The principle of the new handover signalling strategy is to allow the mobile terminal to perform the handover procedure by exchanging handover signalling directly with the new base station. The mechanism used for this purpose must ensure that during the handover process the mobile terminal continues to be in communication with the old base station until a new link is established with the new base station. PRMA was chosen as the medium to implement such a strategy for the following reasons.

(i) PRMA offers superior spectral capacity over the conventional TDMA structure, because a time slot is reserved during activity periods only, as compared to the total period of the call in TDMA. This in effect will give the mobile terminal higher probability of success in attempting to reserve a slot when accessing the new base station.

(ii) The number of slots per frame in PRMA is much greater than that in TDMA, hence allowing the mobile terminal to have more room for contention for a slot; this is illustrated in Figure.(5.16).

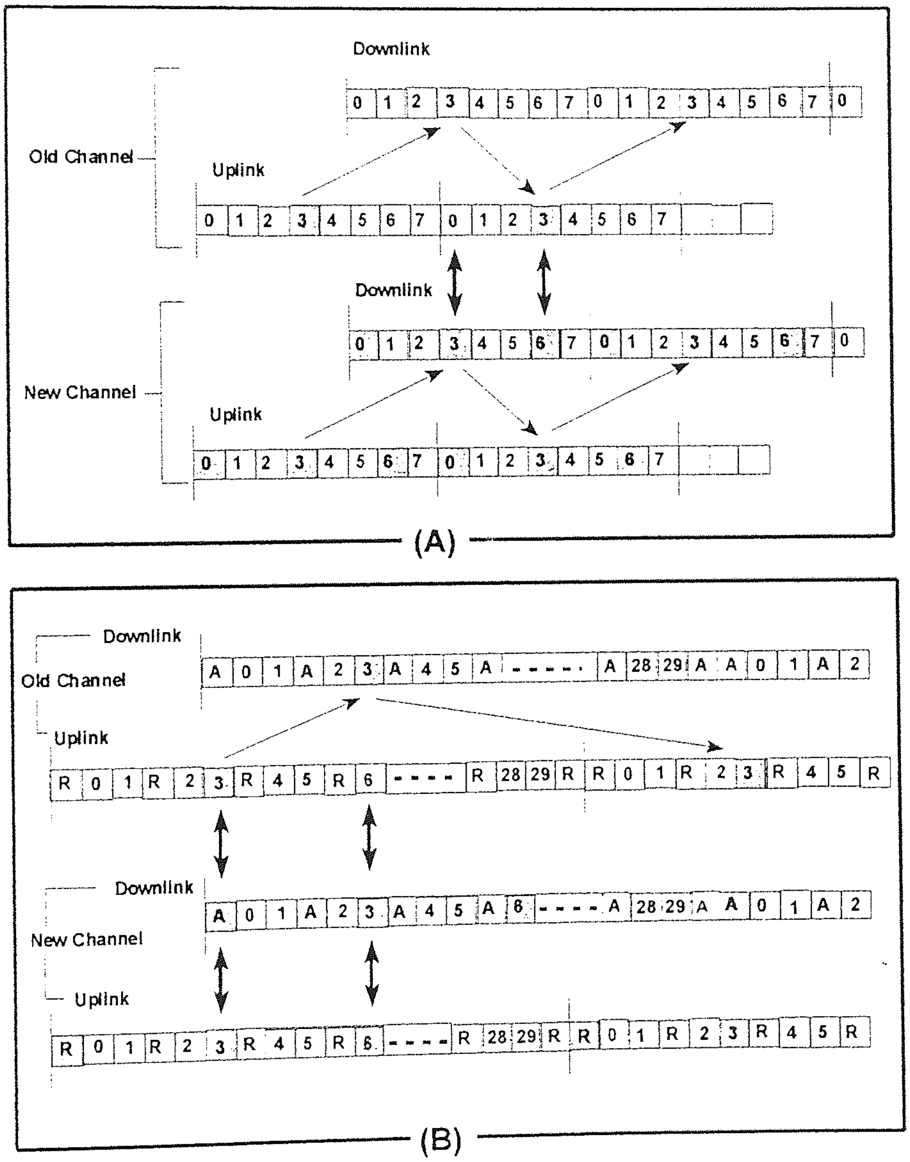


Figure.(5.16): The accessible and inaccessible slots on the new channel for (a) TDMA, and (b) Advanced PRMA.

Figure.(5.16) shows the time slots available for access to a new channel in TDMA and PRMA frames. Since the mobile terminal can communicate with one base station only during a certain time slot, there will be limitations in its period of activity with the new base station. In Figure.(5.16-a) the double arrows indicate the time slots during which the mobile terminal is unable to communicate with the new base station in TDMA. An active mobile terminal occupying slot no. 3 will have only 5 slots available to attempt access on the new channel. Three slots in the new channel are excluded because they coincide with slot no. 3 on the old channel (on the uplink and downlink). It should be noticed that any of the 5 slots can only be accessed provided it is vacant and not already occupied by an ongoing call.

For Advanced PRMA, a mobile terminal attempting to access a channel in a PRMA frame would have to make a reservation on the first R slot available; when the reservation is granted the terminal will be allocated a certain slot for transmission to the base station. In Figure.(5.16-b) the double arrows indicate the activity periods for the mobile terminal in the old channel overlapping with 2 information slots and one R slot in the new channel. This means that the mobile terminal will have 15 R slots available to make a reservation. Once successful, the mobile terminal will have a high probability of being allocated one of the 28 information slots remaining in the PRMA frame. When the mobile terminal accesses a slot in the new channel, it will send its *Handover_Required* message. Upon receiving the Handover_Required message, the new base station will reserve that particular time slot for the mobile terminal to resume its handover signalling. Therefore, there will be no need for the mobile terminal to contend with other terminals for a new slot to resume its handover signalling.

As previously mentioned, the success of the new strategy depends primarily on the spectral efficiency inherent in the PRMA medium access technique, and the greater number of slots per frame in comparison to TDMA. In addition a typical PRMA++ frame will have 72 slots in total instead of 46 slots [101], leading to a higher probability of success in attempting to access the target base stations.

5.6 Results and Performance Comparisons

The results presented in the following sections are split into two parts. The first part lists the performance results obtained from the FCA and DCA schemes simulated in the

TDMA model. While the second part presents the results obtained from the PRMA++ and the Advanced PRMA models and compares their performance with those of TDMA. For each of the models simulated, it was necessary to evaluate different aspects of the system performance and not just concentrate on the handover delay output. Such an overall analysis was necessary to determine the validity and efficiency of the improved handover signalling procedure in comparison to the existing standards.

The TDMA and PRMA simulation models are concerned with complete handover procedures, therefore simulations for the TDMA and the PRMA models were run for 5 minutes instead of 90 seconds, which was the case with the COMNET model shown in section (5.4). The simulations carried out are restricted to voice transmission only. Other considerations, such as bandwidth requirements, have to be accommodated when data transmission is included because of the multiple slot allocation with respect to the application being handed over and the mechanism utilised to allocate the necessary resources.

5.6.1 TDMA Simulations

Handover delays shown in the following results are due to packet transmission delays over the different links in the network and also due to the packet queuing at the network nodes en route to their destinations. Packet processing delays at the network nodes are not included. Handover requests are processed as soon as they are received by the target cell. If there are no free channels then the handover request will be blocked. Blocked handover requests will not be queued to wait for a free channel in the target cell or any other candidate cell.

The traffic loading applied to the TDMA models ranged between 20-160 Erlang per cell. The mobile terminals were uniformly distributed across the network, therefore the results obtained reflected the behaviour of the system in total as well as in the handover cells. DCA schemes adopted utilize the channel re-use provided that a channel satisfies the minimum distance for co-channel interference requirement. It is assumed that when the DCA scheme is in operation, the system is capable of handling the increased number of served terminals due to an increase in the channels per cell ratio. i.e. the system will not be blocking any increase in the spectral capacity.

Figure.(5.17) shows the average number of simultaneous calls in progress for the three different microcell sizes. The FCA supports the lowest number of calls while the 3 different DCA schemes showed better performance than FCA. This is due to the limited number of channels allocated per cell in the FCA scheme which becomes more noticable as the level of teletraffic load increases. The difference between the 3 DCA schemes, DCA for handover requests only (DCA-HO), DCA for handover requests and new call requests (DCA-Call & HO), and DCA for new call requests only (DCA-Call), does not appear clearly until the load reaches around 100 Erlang per cell, where the DCA-HO shows a lower capacity for calls than the other two. DCA-Call supports the highest number of calls while DCA-Call & HO hovers in between the latter two, but more towards the DCA-HO curve.

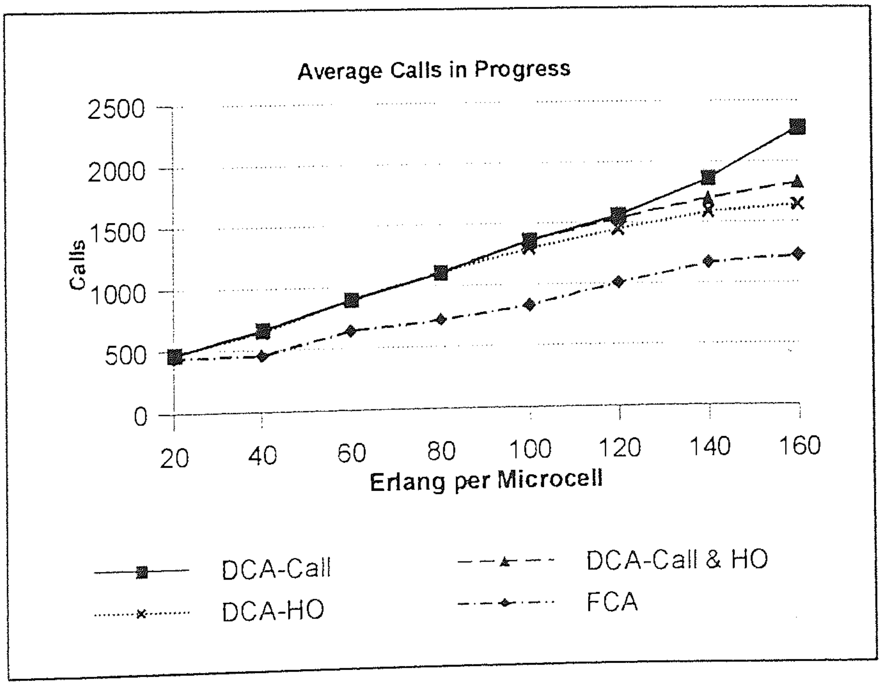


Figure.(5.17): Average simultaneous calls for 30, 40 and 50 metres microcells.

Since the channels in DCA-HO are only allocated to handover requests, a higher probability of successful handovers will be achieved and hence these calls will be continued, thus increasing the number of instantaneous calls supported by the network. Figures (5.18-5.20) show the maximum simultaneous calls for the FCA and DCA schemes in the 30 m, 40 m and 50 m microcell diameters; respectively. It is clear that there are very little differences between the maximum calls in the 3 different microcells, which is as expected because of the limited frequency allocations. Therefore the individual results obtained in the 3 different cells are very close to the average value shown in Figure.(5.17).

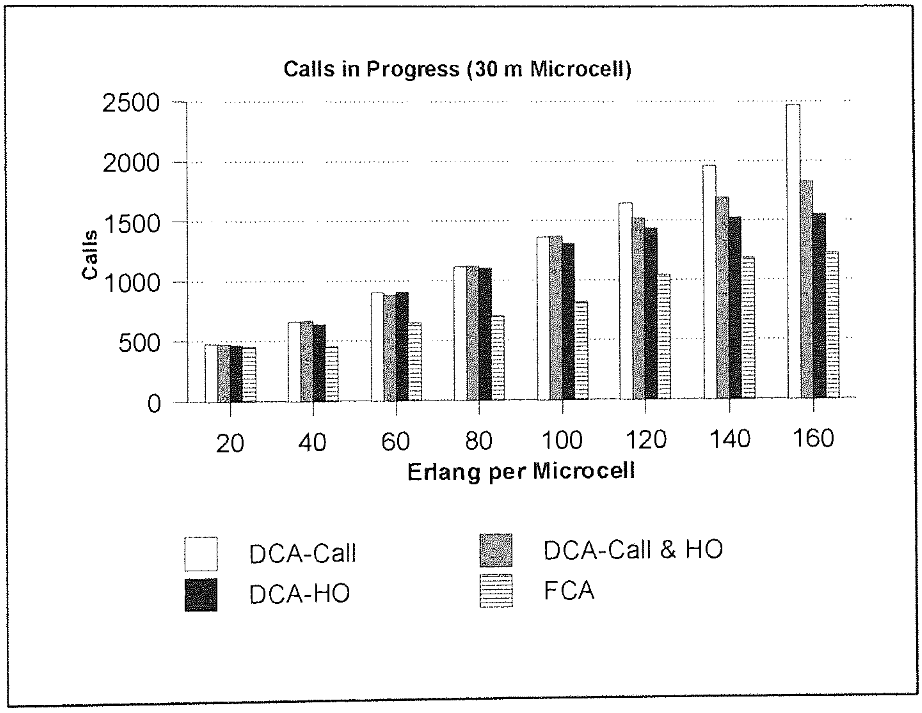


Figure.(5.18): Maximum simultaneous calls for 30 metres microcells.

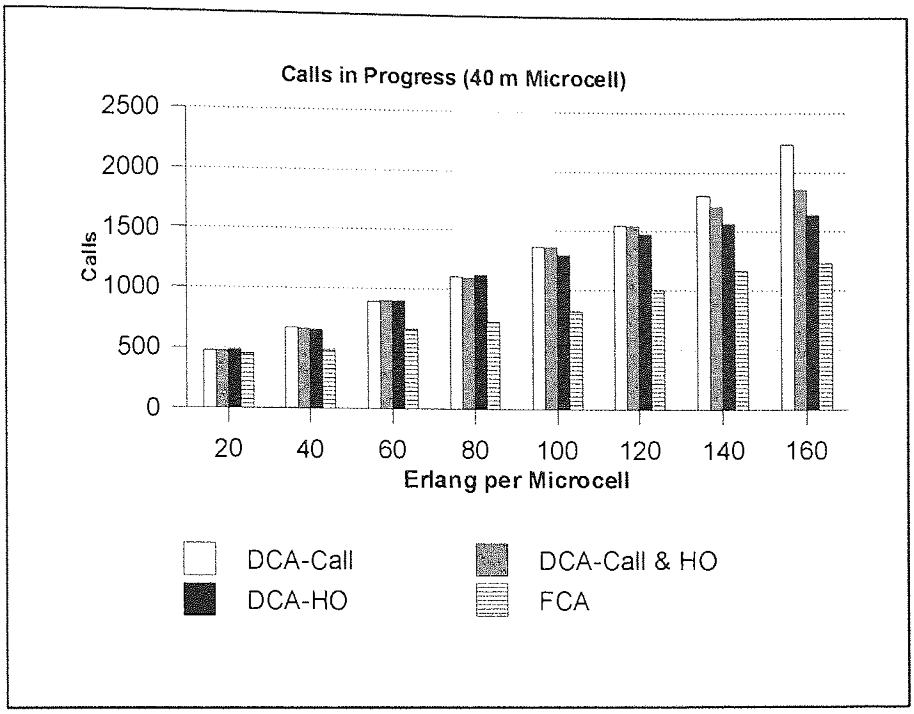


Figure.(5.19): Maximum simultaneous calls for 40 metres microcells.

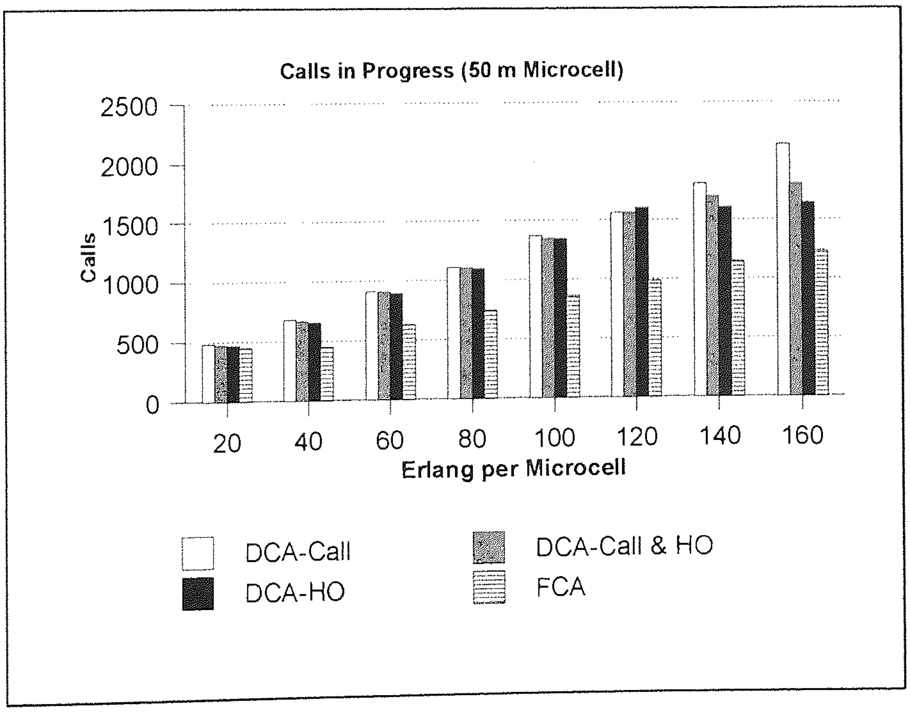


Figure.(5.20): Maximum simultaneous calls for 50 metres microcells.

The average blocking probability of new call requests is shown in Figure.(5.21). The FCA scheme has the highest blocking probability and this is again due to the limited frequency allocation per cell in comparison to the DCA schemes. DCA-HO has a lower blocking probability than FCA but it is closer to the DCA-Call & HO. The best performance is that of the DCA-Call scheme which allocates channels to new call requests only, but this would be at the expense of a higher failure probability for handover requests. The new call blocking probability for each microcell size is shown in Figures (5.22-5.24). Although the results are very close, it is noticable that an increase of 5% to 10% in the blocking probability for each scheme as the cell size increases. This is because an increase in the cell size will result in an increase in the dwelling time a mobile terminal spends in the microcell at a certain speed, and therefore reducing the number of handovers taking place. The longer a mobile terminal remains in the microcell, the longer it will take to vacate the slot it is occupying; this lowers the probability of finding free time slots for new call requests. This is confirmed by Figure.(5.25) which shows the decrease in the average number of successful handovers performed per microcell as the size of the microcell increases.

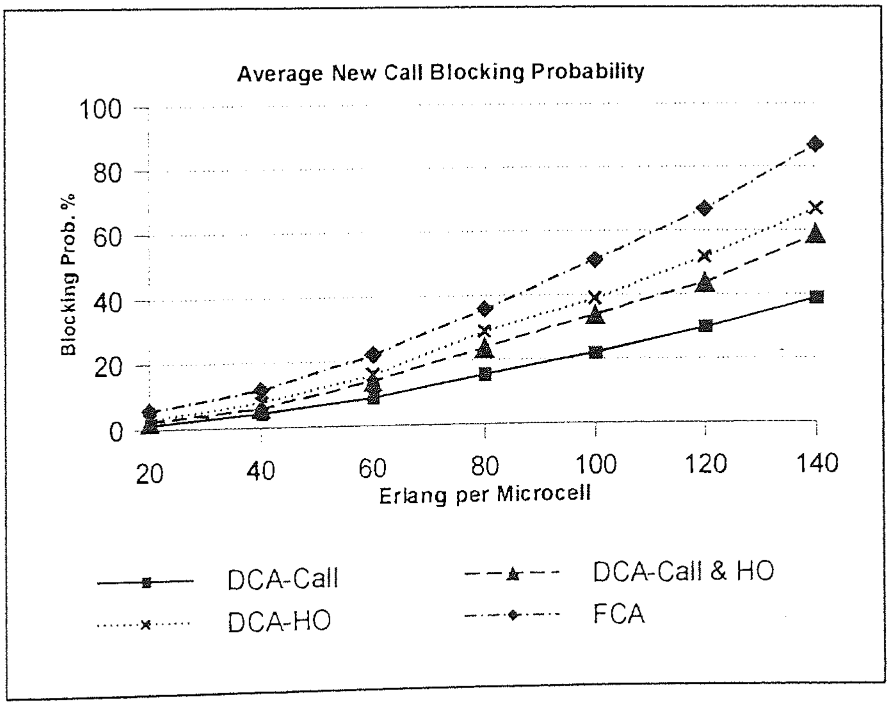


Figure.(5.21): Average new call blocking probabilities for 30, 40 and 50 metres microcells.

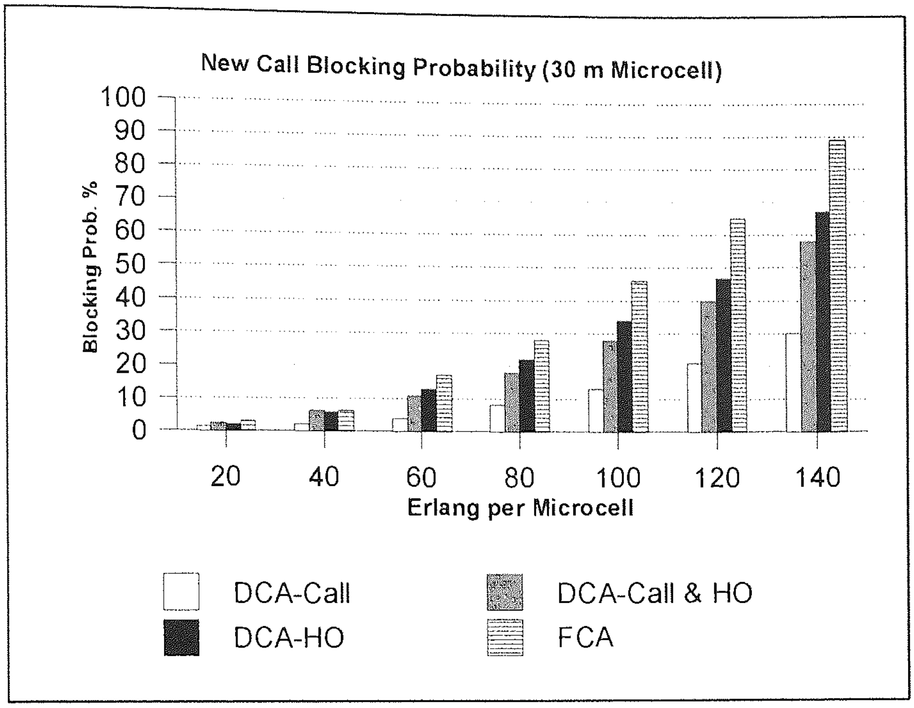


Figure.(5.22): New call blocking probabilities for 30 metres microcells.

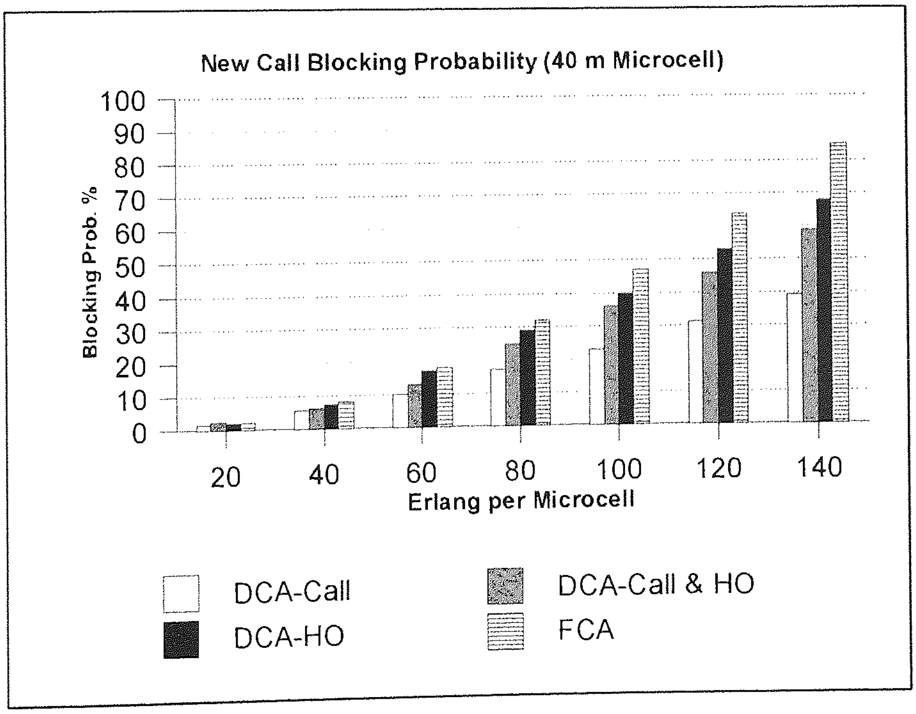


Figure.(5.23): New call blocking probabilities for 40 metres microcells.

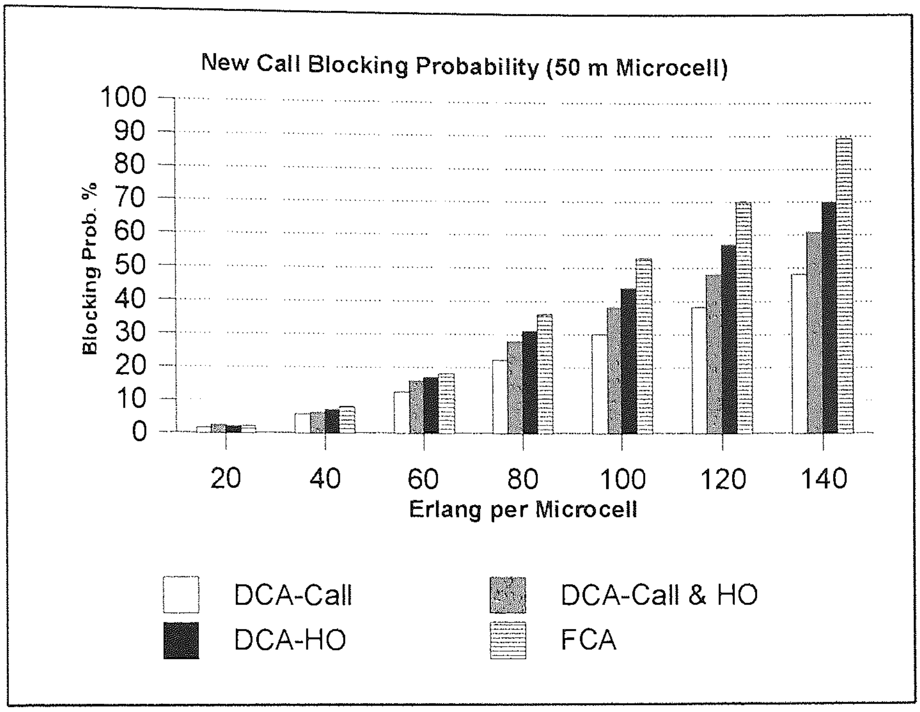


Figure.(5.24): New call blocking probabilities for 50 metres microcells.

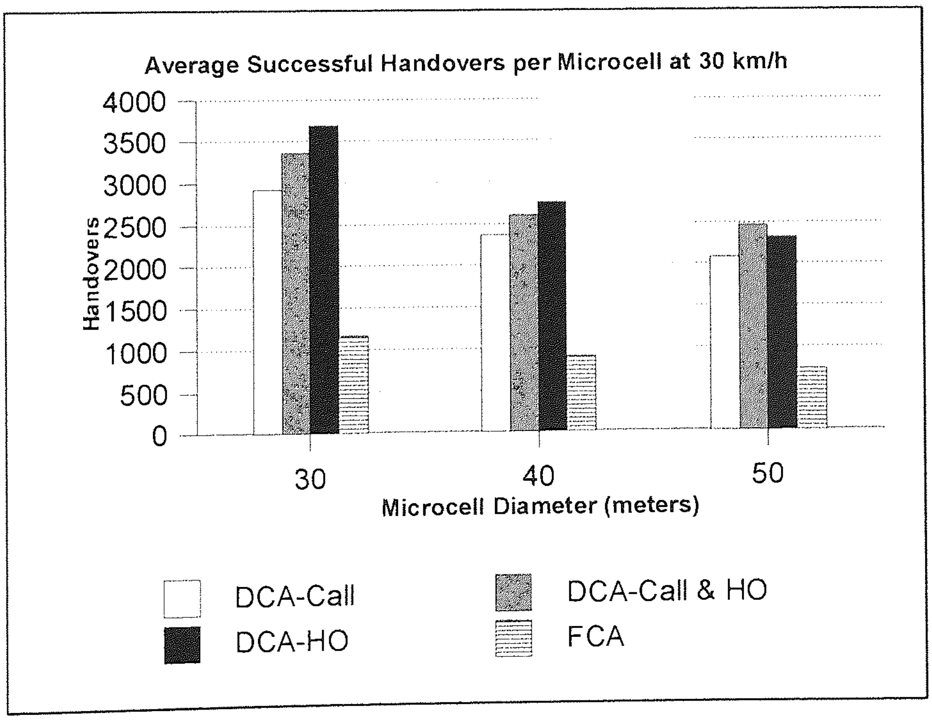


Figure.(5.25): Average successful handovers per microcell at a speed of 30 km/h.

Figure.(5.26) shows the average handover delay in the 3 different microcell sizes. The FCA scheme has the lowest delay figures while DCA-Call has the highest time delay, particularly towards the upper end of the teletraffic loading level. The results for DCA-HO and the DCA-Call & HO exhibit values close to those of the DCA-Call, but the latter two will be lower than DCA-Call as the loading approaches its maximum levels. As explained earlier in section 5.2, the handover delay is greatly dependent on the level of teletraffic handled by the cellular network at the time of the handover and consequently the queuing time required at the different nodes in the network. It can be clearly seen that Figure.(5.26) has a similar trend to Figure.(5.17) which represents the number of simultaneous calls in the system. FCA has the lowest handover delay because this scheme supports the lowest number of simultaneous calls and therefore there is less signalling packets exchanged and less queuing time in the fixed part of the PLMN. DCA-Call exhibits the worst case delays due to the high number of calls being handled by the system. The queuing delays are partly due to the speech packets and partly due to the handover packets from different terminals across the network. It can be clearly seen that the FCA achieves the lowest handover delay but at the expense of higher blocking probability, lower spectral capacity and higher probability of failed handovers.

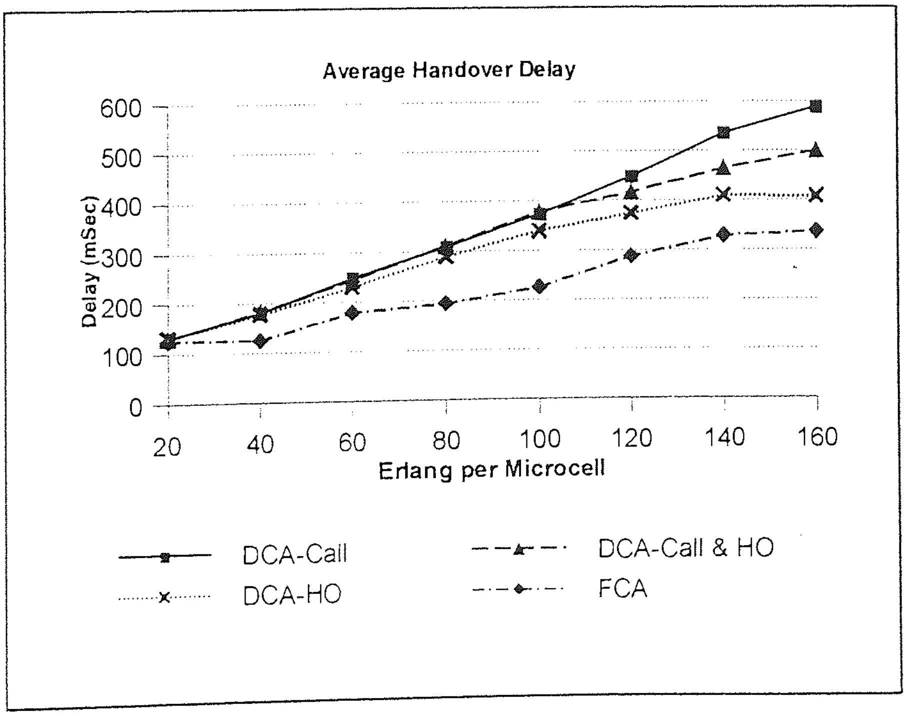


Figure.(5.26): Average handover delay for 30, 40 and 50 metres microcells.

As expected, the number of handovers performed will increase with the speed of the mobile terminal and the success of the handover procedure relies primarily on the availability of a free channel in the target cell. The numbers of terminated (or disconnected) calls due to the unavailability of channels in the target cell are shown in Figure.(5.27), Figure.(5.28) and Figure.(5.29) for 30m, 40m and 50m microcells; respectively. These results are obtained for a loading level of 80 Erlang per microcell. The DCA-HO scheme has the lowest number of failed handovers and that is mainly due to the availability of channels for the handover request. The DCA-Call & HO scheme has a higher number of failures due to the sharing of the channels between new call requests and handover requests. Although FCA demonstrates a high number of failed handovers, the DCA-Call has the highest number of failed handovers and this is due to the large number of calls handled by the DCA-Call scheme.

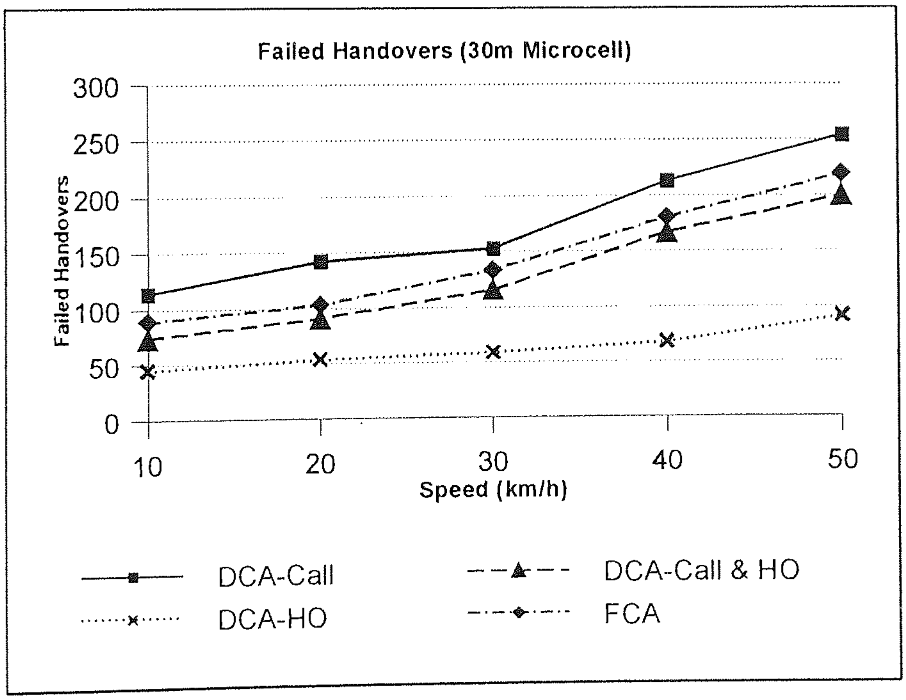


Figure.(5.27): Handover terminated calls in 30 metres microcells.

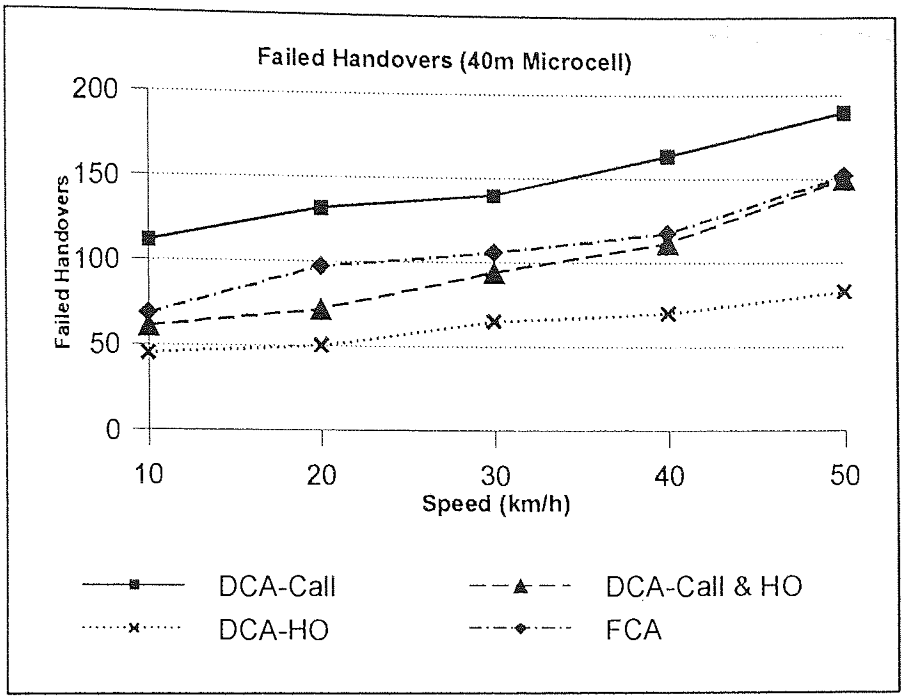


Figure.(5.28): Handover terminated calls in 40 metres microcells.

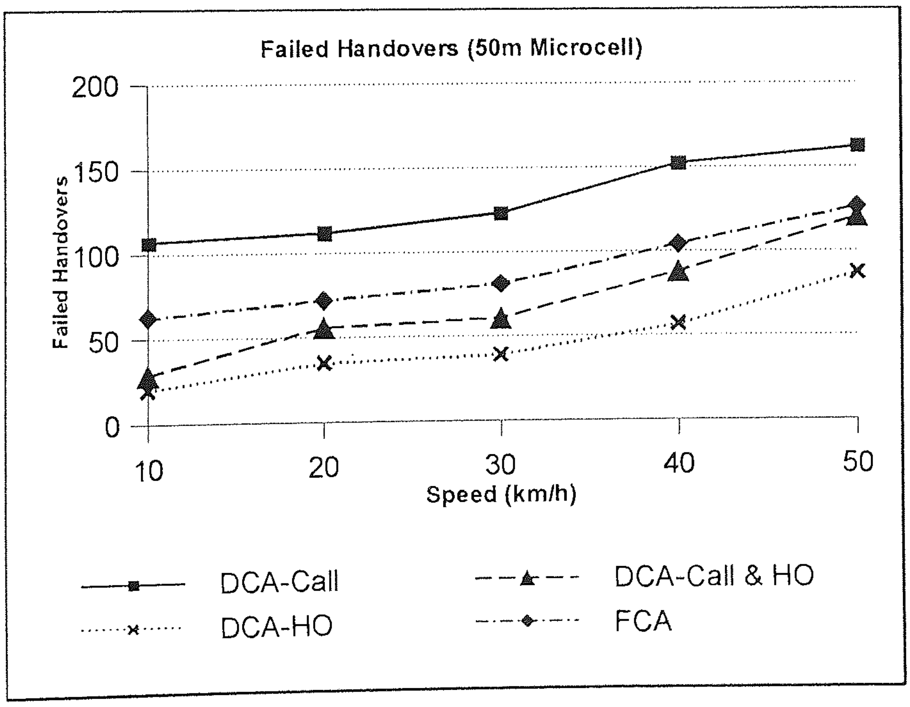


Figure.(5.29): Handover terminated calls in 50 metres microcells.

The numbers of successful handovers performed per microcell with respect to speed are indicated in Figure.(5.30), Figure.(5.31) and Figure.(5.32) for 30 m, 40 m and 50 m microcells; respectively. It is noticeable that the 3 DCA schemes have very similar results, while in the failed handovers figures there is marked difference. Also it is noticeable that the number of successful handovers in the FCA is much less than that in the DCA schemes. This can be explained in Figure.(5.33) which shows the ratio of failed-to-successful handovers. It can be seen that the FCA scheme has the highest ratio and this is due to the fact that FCA supports the lowest number of simultaneous calls in the system, while the ratio for the DCA-Call scheme is much lower because of the high number of simultaneous calls supported by DCA-Call. Despite this difference, DCA-Call is still higher than DCA-Call & HO and DCA-HO because the latter two are designed to accommodate handover requests better than DCA-Call and FCA.

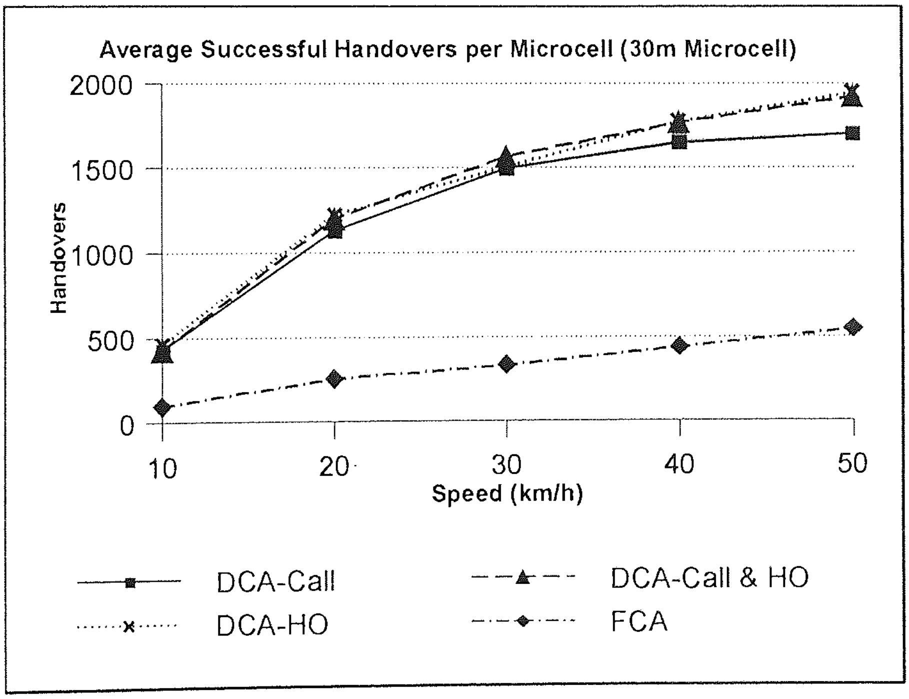


Figure.(5.30): Successful handovers performed in 30 metres microcells.

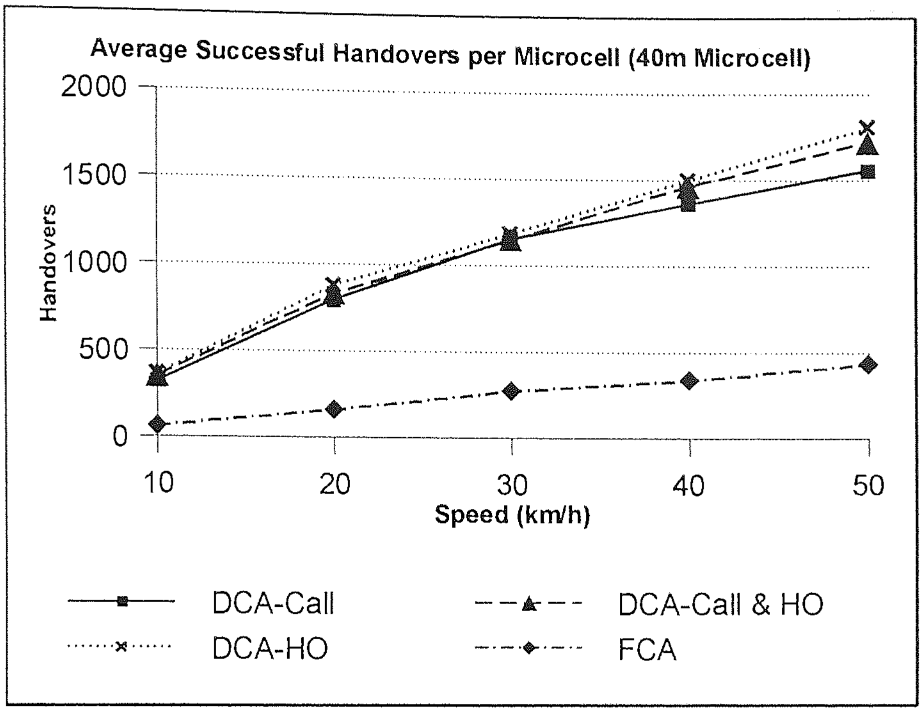


Figure.(5.31): Successful handovers performed in 40 metres microcells.

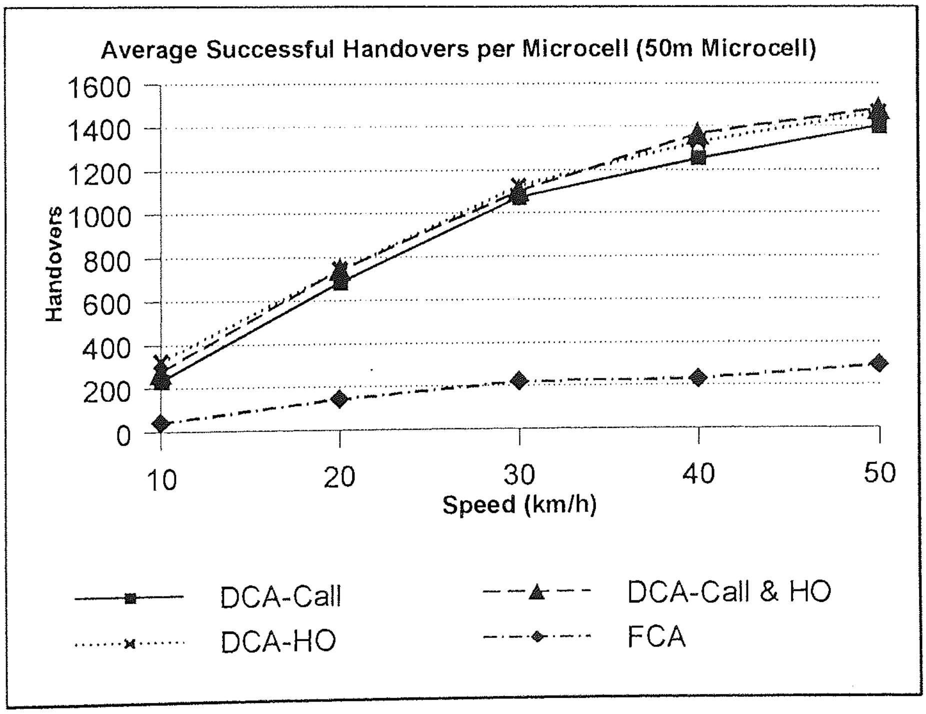


Figure.(5.32): Successful handovers performed in 50 metres microcells.

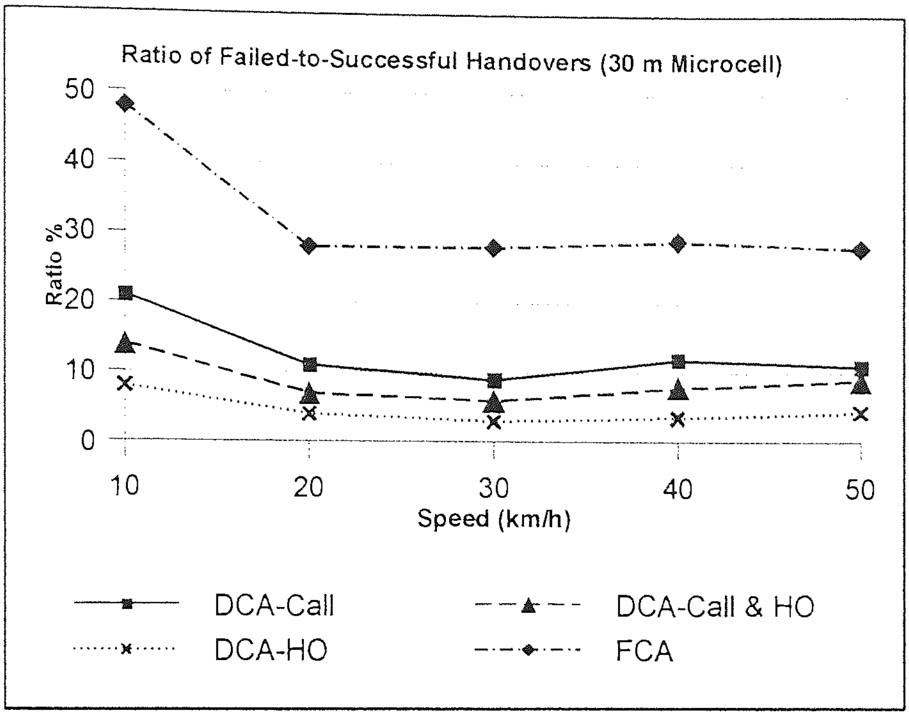


Figure.(5.33): Ratio of failed-to-successful handovers in 30 metres microcell.

It is clear from the results obtained in the TDMA simulation that none of the 4 schemes is capable of providing low handover delay, high spectral efficiency and high handover success probability at the same time. Each one of these schemes suffer from a drawback in one or more aspects which renders it difficult to choose as the optimum scheme. Although it could be argued that the worst handover delays obtained were not higher than the 650 ms limit for successfully completed handovers (section 5.2.3) [95], it is crucial to remember that the handover delays reported in these simulations do not include any processing delays at the network nodes. Also the handover requests are not queued at the destination cell and therefore a handover process will either succeed or fail at its first destination cell. No time was included for the serving cell to interrogate other cells on the candidate cells list. The latter delays are very common in a typical network and they will add to the values already obtained in the above simulations.

5.6.2 PRMA Simulations

The performance metrics for the PRMA medium access protocol are different from those of the TDMA medium access protocol. In TDMA a time slot is dedicated to the call for the whole period of that call and consequently the call would not suffer any interruption. In PRMA, slots are allocated only during voice activity periods, and therefore the ratio of dropped packets to total packets supplied by a voice source determines the overall quality of the call. The packet dropping ratio depends on the access time required for a terminal to obtain a free slot for transmission, which is an increasing function of the number of speech terminals sharing the PRMA channel. Hence, a key measure of PRMA performance in voice transmission is the number of speech terminals that can share a channel within a maximum value of packet dropping probability equivalent to 1% [103]. Also, in TDMA, a new call initiating in a cell will seek a free channel to be used for the whole period of the call. The unavailability of a free channel will result in the call being blocked. Therefore call blocking probability is a performance measure for the spectral capacity in TDMA systems. For PRMA, a new call initiating in a cell will be contending for a free channel (slot) in the same manner and with the same contention probability as the rest of the calls already in progress in that cell. Therefore the call will not be blocked but, instead, its quality will be affected by the access time required to obtain a free slot. Consequently, the system spectral capacity will be indicated by the mean access delay suffered by each terminal.

The teletraffic loading levels for PRMA++ and Advanced PRMA are applied in terms of the number of active terminals per cell instead of call arrival in Erlang. Each terminal would be generating a new call within 2 air frames period (6.44 ms) from the termination time of the last call. The lowest number of mobile terminals per microcell was 60, which is equivalent to the total number of information slots available per microcell. This will give an indication of the behaviour of the PRMA++ and the Advanced PRMA models in the maximum loading situation where the traffic handled is higher than the nominal capacity. The number of terminals per microcell was increased at steps of 20 terminals, reaching a maximum of 140 terminals per microcell. Using equation (5.2) it is possible to show that the teletraffic loading levels applied to the

PRMA simulations are comparable to those applied to the TDMA model. Calls in the PRMA model are generated approximately every 90 seconds per terminal, hence:

λ = mean call arrival rate = 1/call inter-arrival time = 1/90 = 0.011

However, from eq.(5.1) we have $E = \lambda h$, therefore:

Erlang per Terminal = 0.011 x 90 = 1

where h is the mean call holding time.

Hence for 140 terminals per microcell, the total Erlang per microcell = 140 E, which is 20 Erlangs less than the maximum loading level for the TDMA simulations.

For inter-BSC handover, two different procedures were investigated for the PRMA++ and the Advanced PRMA. In the PRMA++, when the serving base station detects a need for handover it will issue a *Handover_Required* message to the target base station via the fixed network requesting a handover slot. The target base station would reserve a certain time slot especially for the mobile terminal to perform its handover signalling, the chosen time slot would be an already vacant slot. If no free slots are available then the target base station will cancel one of the ongoing call reservations and allocate its slot to the handover terminal. The signalling exchanged following this would be the same as in the GSM handover procedure (section 5.2.3). It can be said that the typical GSM handover procedure is followed in PRMA++, apart from one major difference, which is the lack of the candidate base stations list. Because, in a packet access mechanism, a slot is only allocated to the call in voice activity periods, the target base station will always be capable of accommodating a handover request and consequently there is no need for a list of candidate base stations.

In the Advanced PRMA, the handover procedure adopted is the same as the improved handover procedure explained earlier in Section 5.3.1. When the mobile terminal detects the need for handover it will attempt to access a time slot in the target base station by contending with other terminals already in the target cell. Once the mobile terminal acquires a slot, it will send the *Handover_Required* message to the target base station. The slot occupied by the handover terminal will be reserved for the handover terminal until the handover related signalling is exchanged and the call is re-routed to the new base station. As previously mentioned in section 5.3.1, the improved handover procedure bypasses the fixed network and assigns greater functional responsibility to the mobile terminal.

Unlike the TDMA simulations, one cell size and one speed were chosen for all the PRMA++ and Advanced PRMA simulations. Cell diameter was 50 m and the mobile terminal speed was 50 km/h. Neither the speed nor the cell diameter were changed due to the following reasons:

(i) Changing the terminal speed or the cell size will only affect the number of handovers taking place; it was shown earlier that it does not affect the handover delay.

(ii) In the handover procedures of both PRMA++ and Advanced PRMA, a handover request will always be allocated a channel (either by the target base station or by contention, respectively), therefore there will not be any blocked or failed handovers and all handovers are considered successful.

In addition to the above, the fact that no dynamic channel allocation was implemented in the in PRMA medium access technique means both PRMA++ and Advanced PRMA will have a similar performance regarding the number of handovers performed.

Figure.(5.34) shows the handover delay for the PRMA++ and the Advanced PRMA handover procedures. Clearly, the delay figures for handover signalling via the fixed network are much higher than those for the handover signalling via the air interface. This is because handover delay in the PRMA++ is greatly affected by the number of simultaneous calls handled by the fixed part of the PLMN, which leads to higher queuing delays at the network nodes (similar problem to the TDMA model). For the Advanced PRMA, the handover delay is mainly due to the access time required by the mobile terminal to reserve a slot in the target cell. Also, the PRMA++ handover signalling requires the exchange of eight messages to carry out a complete handover procedure, while in the Advanced PRMA scheme the handover procedure requires the exchange of three messages only, and therefore performing faster handovers.

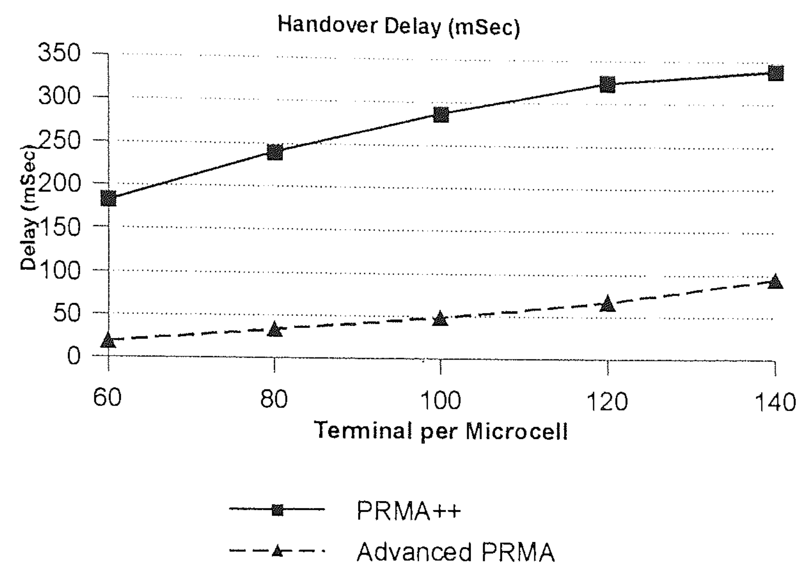


Figure.(5.34): Handover delay for the PRMA++ and the Advanced PRMA handover procedures.

Achieving faster handovers also helps in lowering the mean access delay in the system. This is because a mobile terminal during handover would be occupying two channels to communicate with the old and the new base stations simultaneously. Therefore faster handover ensures quicker release of a slot with the old base station. This can be seen clearly in Figure.(5.35) showing the mean access delay for the PRMA++ and the Advanced PRMA models. Advanced PRMA demonstrates a better performance than that of the PRMA++ by approximately 0.5 ms for most of traffic loading levels. Also, considering the packet dropping threshold of 2 air frames period (2x3.22 ms = 6.44 ms), we can see clearly that Advanced PRMA is capable of supporting around 110 terminals within the acceptable packet dropping probability, while PRMA++ is only capable of handling 80 terminals within the same acceptable limits.

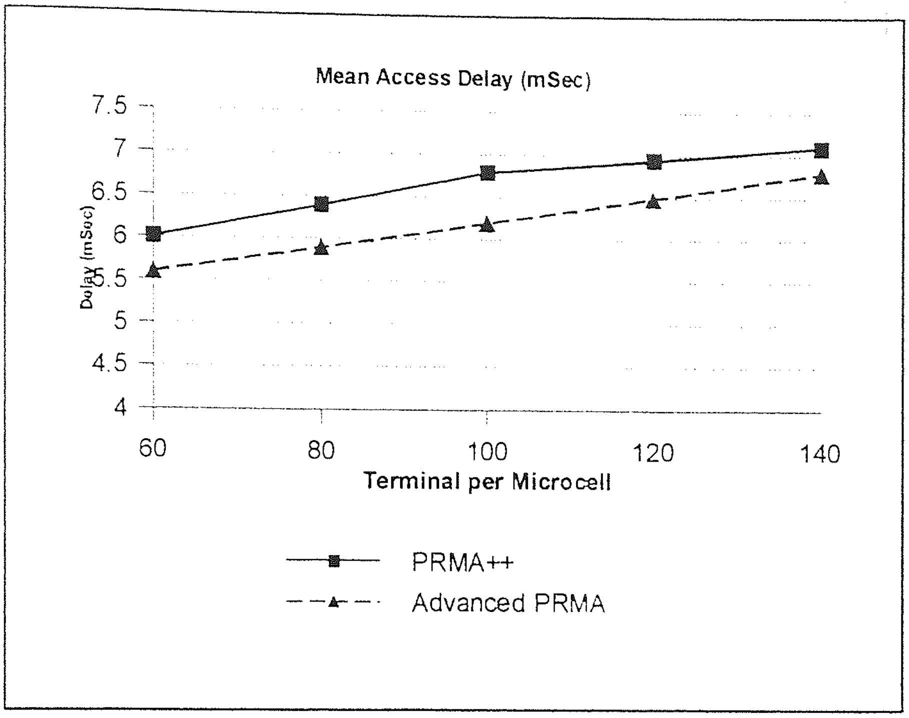


Figure.(5.35): Mean Access Delay.

Figure.(5.36) shows the calls with higher than 1% packet dropping probability, while Figure.(5.37) shows the average dropped packets. In both figures the Advanced PRMA has a slightly better performance because of the low mean access delay as compared to that of the PRMA++. It is obvious that the reduced handover signalling for the improved signalling procedure has a substantial effect in reducing the mean access delay even though the mobile terminal has to contend for a time slot to start the handover signalling. Also it should be mentioned that a reduction in the number of reservation slots would result in delays higher than those already obtained with 16 reservation slots.

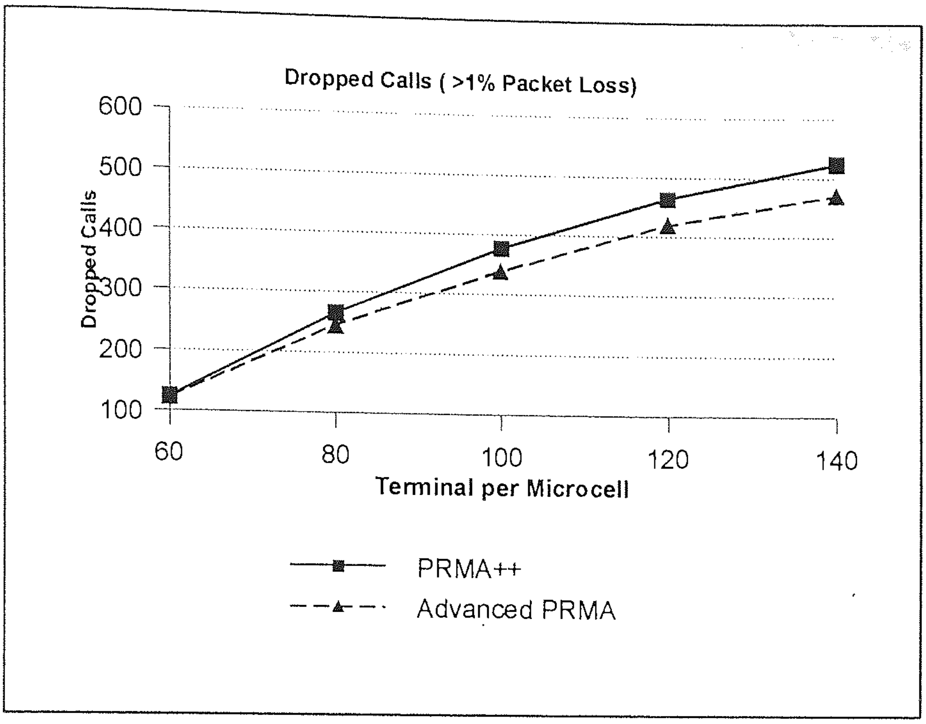


Figure.(5.36): Calls with >1% packet dropping probability.

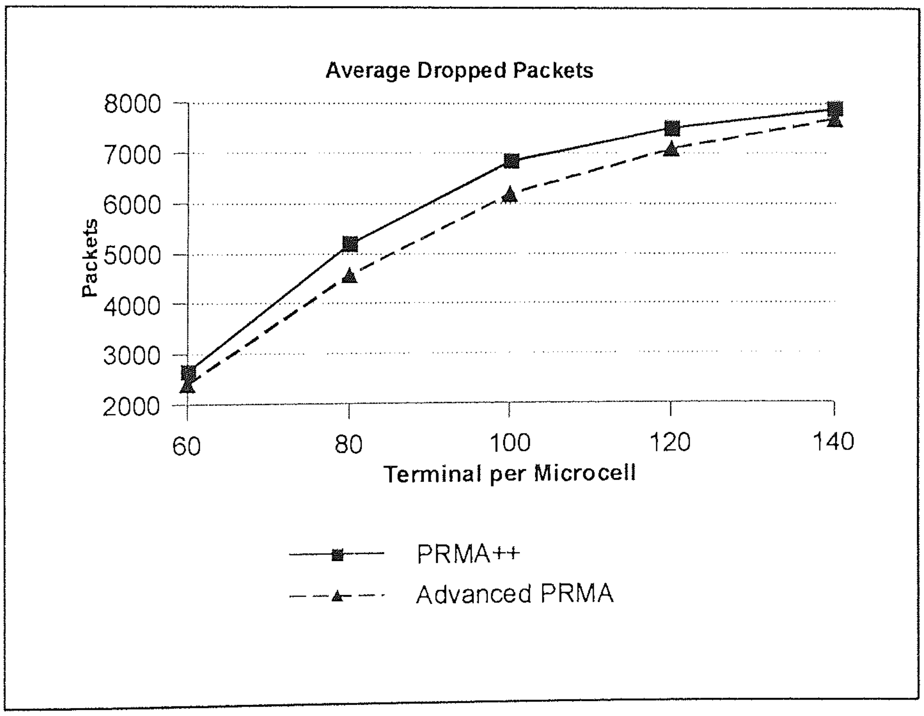


Figure.(5.37): Average dropped packets in PRMA++ and Advanced PRMA.

Due to the changing speech pattern for each user during a call, the number of packets comprising a speech burst could vary depending on the level of voice activity for each user. The results shown so far have presented the PRMA++ and the Advanced PRMA handover performance for voice transmission with average burst length of approximately 30 packets. To demonstrate the effect of the speech burst length on the performance of the PRMA++ and Advanced PRMA, the two models were simulated with speech bursts lengths ranging from 10 to 45 packets. Figure.(5.38) and Figure.(5.39) show the effect of the speech burst length on the handover delay in the PRMA++ and Advanced PRMA; respectively.

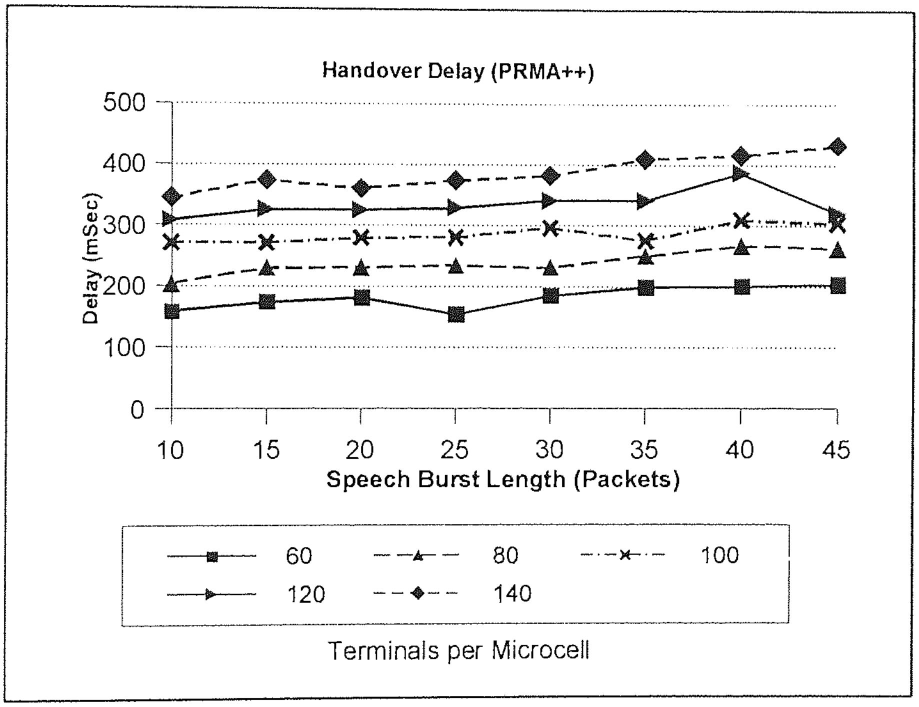


Figure.(5.38): Handover delay vs. speech burst length for PRMA++.

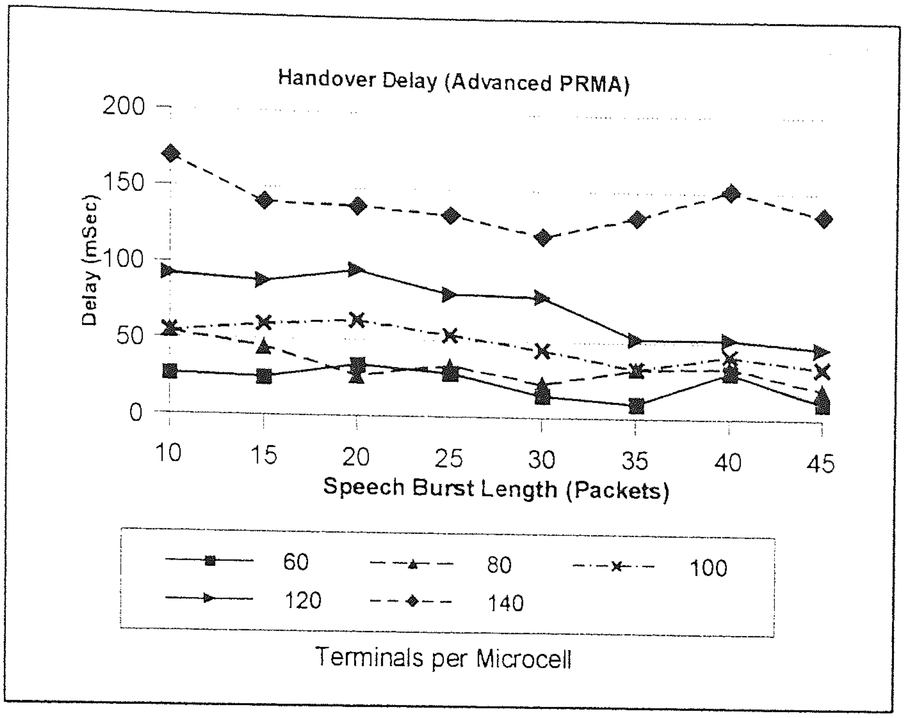


Figure.(5.39): Handover delay vs. speech burst length for Advanced PRMA.

Figure.(5.38) shows that for certain number of terminals per microcell, an increase in the number of burst packets will result in a slight increase in the handover delay. Also changing the number of terminals per microcell will result in relatively equal steps of change in the handover delay. The case is rather different with the Advanced PRMA as handover delay becomes lower with the increase of burst packets and the difference between handover delay for different loading levels (terminals per microcell) increases as the number of terminals increase. An explanation for the above results is presented in the following paragraph.

When a mobile terminal already has a reserved time slot, an increase in the number of packets in a speech burst means that the terminal will be reserving the time slot for longer periods. If the majority of terminals in a cell would have long speech bursts, this would result in more terminals holding on to their time slots and, consequently, there will be fewer contending terminals for the remaining free slots. The remaining terminals will have higher probability of success in acquiring a free slot provided that the number of terminals per cell does not increase. Therefore the increase in the number of packets per burst would have a stabilizing effect on the packet access mechanism and reduce the number of contention events taking place.

In principle, both PRMA++ and Advanced PRMA should show similar handover delay figures when the burst length is increased. However, it is only possible to see it in the Advanced PRMA because the handover delay is dependent mainly on the access time delay. For the PRMA++, the access delay figures are over-shadowed by the queuing delay in the fixed part of the PLMN, resulting from the increasing numbers of simultaneous calls handled by the network. The above can be confirmed by looking at the mean access delay for the PRMA++ and the Advanced PRMA shown in Figure.(5.40) and Figure.(5.41); respectively. It is clear that for the different numbers of terminals per microcell, the mean access delay decreases with the increase in the number of packets, even though the reduction might not be in similar proportions. Also, it is important not to confuse the declining mean access delay values shown in Figure.(5.40) and Figure.(5.41) with those presented earlier in Figure.(5.35), This is because the latter represents the average values for delays presented in Figure.(5.40) and Figure.(5.41).

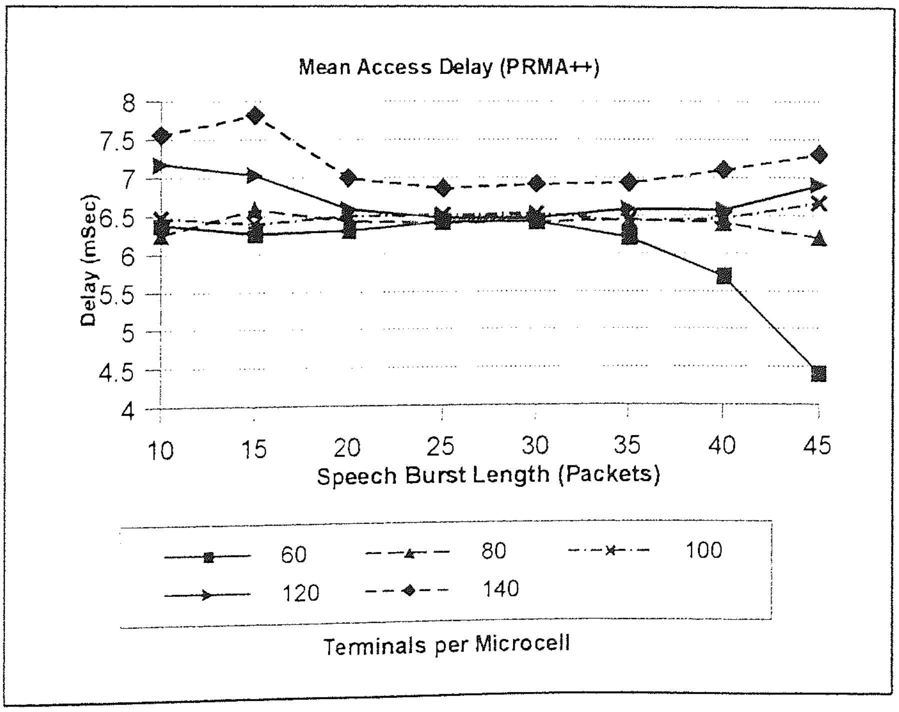


Figure.(5.40): Mean access delay vs. speech burst length for PRMA++.

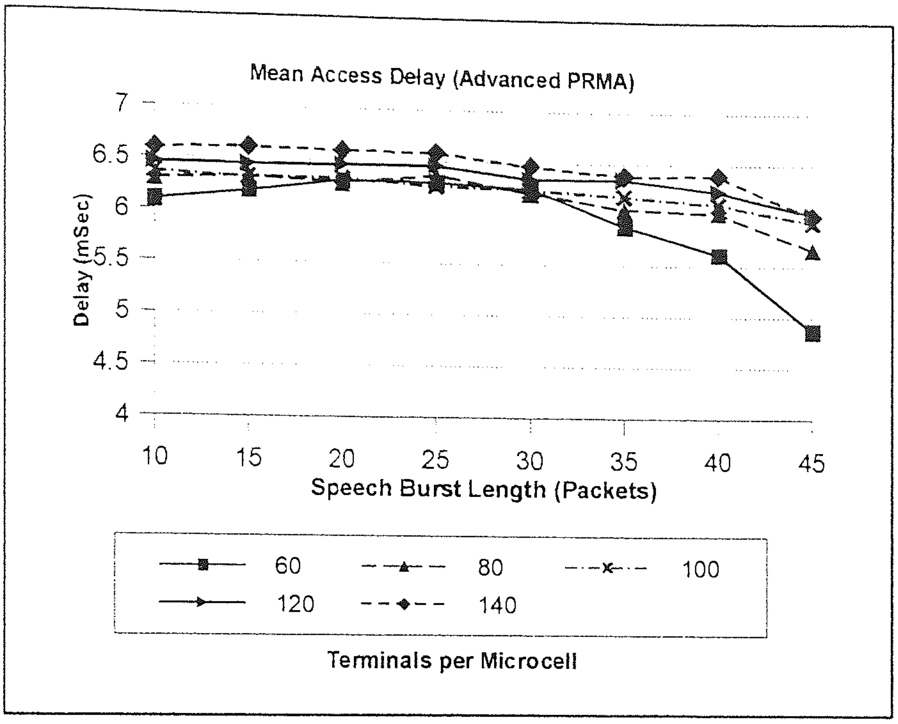


Figure.(5.41): Mean access delay vs. speech burst length for Advanced PRMA.

Due to the inherent nature of the human speech pattern, the speech bursts will continuously have changing lengths and the final effect is the average of all the lengths. Hence, it can be said that changing the burst lengths will not be highly effective for voice calls. However, burst length variation can be better exploited in data transmission. This is because for a certain transmission rate a data application would regularly transmit data blocks of similar sizes. Hence, in cases where a number of mobile terminals are engaged in data transmission, a central control mechanism would be implemented to alter the data block sizes for each application to ensure optimized handover and mean access delay performance as shown earlier in the PRMA++ and the Advanced PRMA results. Such a control mechanism should introduce changes within the allowed bandwidth and delay limits for each data application. Table.(5.6) below shows a performance comparsion in terms of the number of frames lost when transmitting a 300 frames video sequence equivalent to 147.5 Mb. Different transmission rates have been considered to represent different resolutions [107]. The handover delay figures adopted are the values obtained from Figure.(5.34) at a load of 60 terminals per microcell.

Transmission Rate	*36.5 Mb/s <==============> 9.1 Mb/s*			
Compression Rate	1000	10 000	1000	10 000
Time to Transmit 300 frames	4.04 ms	0.404 ms	16.2 ms	1.62 ms
Frames lost at handover delay of 25 ms with Advanced PRMA	6.1 x 300	61 x 300	1.5 x 300	15.43 x 300
Frames lost at handover delay of 175 ms with PRMA++	43.31 x 300	433.1 x 300	10 x 300	108 x 300

Table.(5.6): Time delay and frame loss during handover using PRMA++ and Advanced PRMA

Comparing the output from the Advanced PRMA to that from the FCA, DCA and PRMA++, a considerable reduction in the handover delay has been achieved by diverting all the handover signalling from the fixed network and transferring it to the air interface. However, because the handover delay in the Advanced PRMA is dependent on the mean access delay, it can be said that the handover delay is indirectly dependant on the level of teletraffic handled by the network. Also, the reduction in the handover delay results from the combination of air interface signalling and the new functional role assigned to the mobile terminal. It is important to notice that the improvement in the handover delay did not cause any deterioration in the mean access delay and the average dropped packets. Instead there was an improvement in comparison to the performance of the PRMA++.

5.7 Conclusion

The new handover signalling strategy is aimed at reducing the total handover delay and improving the handover procedure by redefining the functional roles for the base station and the mobile terminal. A number of functional responsibilities were transferred from the base station to the mobile terminal to allow the decentralization of handover management. This is clearly represented by making the mobile terminal capable of initiating the handover procedure instead of relying on the base station to perform this crucial task. Such a change in the functional entities required the introduction of a new handover signalling sequence different from that currently implemented in the GSM standard.

The results obtained showed that for the FCA, DCA-Call, DCA-HO, and DCA-Call & HO schemes, the handover delay is greatly dependent on the traffic handled by the network. The FCA had the lowest handover delay, but it suffered from the worst spectral efficiency and this is due to the limited resources of the fixed channel allocation scheme. The three DCA schemes demonstrated close results but the increase in the simultaneous call traffic led to an increase in the handover delay; it also resulted in an increase in the number of failed handovers. The number of handovers taking place increases with the speed of the mobile terminal and the reduction in the microcell size. However, handovers in TDMA were more sensitive to the latter factors than was the case with PRMA. This was due to the limited spectral capacity and the lack of voice activity allocations in TDMA. In PRMA, handovers had a higher probability of success due to the allocation of slots during activity periods. The Advanced PRMA demonstrated the effect of implementing the new handover strategy on the performance of the PRMA++ scheme. Handover delay was greatly reduced while the mean access delay and packet dropping probability were maintained at levels similar, and in some cases better, than those of the PRMA++.

The maximum handover delays obtained in this chapter are within the acceptable limits for the GSM standard normal operation. However, these delays are only due to queuing at the fixed network nodes. Delays due to packet processing at the network nodes were not included in these simulations. This is because such delays are dependent on the type and speed of the equipment used which would differ from one network operator another. Also, the simulations assumed that there is only one target cell for each terminal performing a handover, hence there were no delays included for interrogating other cells on the candidate cell list. The latter delay should be considered in cases where the mobile terminal is capable of receiving transmission from more than one cell in the handover direction, e.g. microcells and macrocells in outdoor urban areas. Therefore, it is vital to remember that in a real network, the delay figures obtained could rise to levels beyond the GSM acceptable limits, and consequently increase the number of failures in the handovers carried out. Lowering the minimum levels for handover delay would increase the probability of handover success. It allows the handover procedure to undergo further delays before reaching the maximum limits, hence increasing the margin of success. This is quite effective for future applications requiring lower thresholds for the handover delay and more stringent transmission delay requirements. As already stated, the improvements obtained from the new strategy have resulted from a combination of signalling path re-routing and a major change in the functional structure within the PLMN elements.

CHAPTER SIX

DISCUSSION, CONCLUSION and FUTURE WORK RECOMMENDATIONS

6.1 Results Discussion

In the GSM system, the base station initiates a handover procedure when detecting a drop in the perceived quality of the uplink transmission from a mobile terminal. The serving base station will interrogate the candidate base stations (reported by the mobile terminal) to determine the availability of a free channel. Until a suitable channel has been found, the handover procedure signalling between the old and the candidate base station(s) is exchanged via the fixed part of the PLMN. However, signalling packets exchanged via the PLMN can experience delays due to packet queuing and congestion at the different network nodes. Such delays can increase with an increase in the number of mobile terminals simultaneously supported by the PLMN. For intra-BSC handovers the signalling is exchanged within the BSC and the handover process is relatively fast. However in the inter-BSC handovers, the signalling has to be exchanged via the MSC which would impose extra delays, particularly when more than one candidate base station has to be contacted to find a free channel.

The new handover strategy introduces two concepts: (i) the handover signalling is diverted via the air interface; this avoids the fixed part of the PLMN and any delays or congestion that might occur due to the traffic levels. (ii) the control of initiating the handover procedure from the base station is transferred to the mobile terminal which enables the reduction in the number of messages exchanged to perform a complete handover. The latter is particularly beneficial in dealing with the 'corner-effect' where a mobile terminal turns round a corner and the levels of uplink and downlink transmission can deteriorate rapidly. The fact that the mobile terminal is capable of communicating directly with the new base station means that it can initiate a handover procedure as soon as it detects a drop in the downlink transmission of its old base station, and start accessing a channel in the next base station with a stronger transmission. This is especially effective when the mobile terminal is moving in a microcells-only environment and there are no overlaying macrocells that the terminal can attempt to access.

In the new strategy, the mobile terminal will carry out measurements on adjacent base stations transmission, but instead of reporting a list to the serving base station as in GSM, it will only update its register with the identity code of the strongest base station. Implementing PRMA as a medium access technique meant that during handover the mobile terminal is capable of accessing the target cell channels by using the inherited statistical multiplexing in the PRMA mechanism to contend for a free time-slot. Since the mobile terminal is capable of communicating with and accessing the target base station, the new strategy had shown that the mobile terminal did not require a list of candidate base stations, which is the case in the GSM standard. In addition, the delay that the mobile terminal will have to endure during the handover procedure in PRMA is the contention delay to access a free slot. Once a slot is acquired, it will be retained until the handover procedure is complete. Results obtained in Chapter Five have shown that the slot access delay is considerably lower than the queuing, congestion and multiple cell communication delays which takes place in the fixed part of the PLMN.

Chapter Five shows simulation comparisons between FCA scheme and three different DCA schemes using TDMA as a medium access technique. The simulated DCA schemes included DCA for handover requests, DCA for new call requests, and shared DCA between the latter two provided that channels were allocated within the co-channel interference requirements. The results obtained showed that despite the increase in the number of successful handovers in the case of DCA for handover requests, handover delays were affected by the teletraffic levels in the PLMN even though they were slightly lower than the rest of the DCA schemes. Whereas in the case of DCA for call requests, an increased number of simultaneously supported calls have resulted in higher handover delays and less successful handover attempts. Allocating channels dynamically to both call and handover requests provided a compromise between the two previous schemes but not a major improvement in the overall performance, particularly with the handover delay. FCA demonstrated lower delay figures than those for the DCA schemes but that was at the expense of reduced spectral efficiency and a lower number of successful handover attempts.

In comparison with the above mentioned TDMA based channel allocation schemes, the new handover strategy demonstrates a substantial reduction in the time delay required for performing a complete handover procedure. Even when comparing the Advanced PRMA with the conventional PRMA performance (in which handover procedure is carried out via the PLMN), the Advanced PRMA demonstrated a considerable improvement in the handover delay figures. The Advanced PRMA demonstrates a similar access delay and spectral capacity characteristics to those of the conventional PRMA.

6.2 Conclusions

Previous research work on improving handover performance has investigated numerous techniques for dealing with queuing and priority schemes. Such schemes included queuing handover requests at the new base station as long as the mobile terminal is residing in the overlap area between the old and the new cell [22,38]. Selective handover methods have also been considered in which certain calls are handed over to adjacent cells when the number of calls handled by one cell exceeds certain thresholds [31]. Executing queued handover requests according to the deterioration in the uplink transmission as perceived by the old base station was also suggested [30]. The disadvantage of such schemes is the additional complexity involved in updating the queuing lists whether it was on a network level or on a base station level. Also, the majority of these schemes are based on the assumption that there will always be a sufficient overlap area between the adjacent cells, something which might not be a common occurrence in all microcellular environments.

Reserving certain channels solely for handover requests while the rest of channels in the cell are shared between new originating calls and handover requests has also been considered as a method to improve handover performance [22,27,37,91]. Reserving channels for handover performs efficiently in low to moderate loads, but in high loads the system will be saturated and achieving a successful handover will not necessarily mean a continuation of the call, because the channel would be reserved for the handover process only [37]. In addition, reserving sets of channels will result in higher rates of new call blocking within the cell. DCA has also been implemented to assign channels to handover requests as well as new call requests [85,86]. However, this kind of channel allocation would add a new signalling burden to the network infrastructure especially when implemented on a large scale. Additional information processing will be necessary to determine which channels can be used within the co-channel interference requirements [87].

Multiple hierarchical systems with microcells, macrocells, and a mobile satellite segment have been investigated as a method to improve the probability of handover success and achieve global coverage [40,41,47,25]. However, switching the call between the two layers would be subjected to a number of parameters such as the speed of the terminal, the cost of the link and the spectral capacity available at both terrestrial and satellite levels [43]. The handover process to and from the mobile satellite system is more complicated than it is within terrestrial systems, this is due to the motion of the mobile terminal and the satellite beam spot. Although the use of satellite and overlying macrocells would present a perfect solution for the handover problem in dense microcells, more research efforts are required to evaluate the effect of teletraffic levels on handover performance in isolated microcellular environments [104]. Although the microcellular environment might represent a specific area in the cellular coverage map, smaller cells will constitute a considerable proportion of the future cellular networks, and the performance in the microcells will affect the overall performance of the cellular network.

The majority of past research work is concerned with manipulating the available frequency spectrum or integrating more than one system to improve the handover success probability. However, it is evident that all the suggested policies and systems utilise the fixed network in performing any signalling related to the handover procedure. Hence, the performance of these systems is greatly affected by the levels of call traffic supported by the network, and consequently any cellular system operating at its maximum capacity will not be capable of providing an acceptable handover performance. The research presented in this thesis was thus aimed at:

(i) reducing the total handover delay for mobile terminals roaming in microcellular environments without overlaying macrocells,

(ii) minimising the effect of the overall system traffic on the efficiency and the time delay of the handover procedure, particularly for mobile terminals situated on mobile platforms.

These goals where achieved by diverting the handover signalling from the fixed part of the PLMN to the air interface using the PRMA technique as a medium access protocol. The new strategy involves the decentralisation of the handover procedure control by transferring the handover initiation function to the mobile side. This approach reduces the number of exchanged signalling operations and hence speeds up the handover process.

It should be clear that the objective of this research was to improve the handover performance of the TDMA-based GSM standard by re-routing the handover signalling through the air interface. The objective was not to provide a comparison of the handover performance of the major multiple access techniques like TDMA, FDMA, CDMA, and PRMA. However, it was the inherited characteristics of the reservation based packet access mechanism and the functionality and control requirements of the new signalling strategy that made PRMA a suitable mechanism for performing the improved handover procedure.

6.3 Future Work Recommendations

In addition to the conventional voice communications, the Third Generation mobile communication systems are expected to provide a wide range of non-voice based applications like Data services (e.g. file transfer and Internet access) and Multimedia applications (Video conferencing and image transmission) [105,106]. Also it is expected to provide other services which are mobility specific, such as the capability of maintaining the communication while roaming across different cellular networks with different standards around the globe. Efficient implementation of these services will require reliable handover techniques to maintain the expected level of quality of service, while the mobile terminal is moving within the same network or across different networks. The handover strategy suggested in this thesis represents an effective policy to provide fast and swift handover procedures, particularly for voice applications as indicated in the results in Chapter Five. However, applying this strategy to future Third Generation systems would require the undertaking of a number of practical steps before the strategy could be applied successfully. These can be summarised in the following points:

- Multimedia services are based on the transmission of synchronised audio, real-time images, and data between two or more parties, with the addition and deletion of resources and participants within a single communication (e.g. conferencing services). Complex compression techniques are required to enable the transmission of the vast amounts of visual information over the limited bandwidth of the mobile channel [107]. Providing these applications to mobile terminals moving at speeds through the cellular coverage area would require robust handover techniques to reduce delays that might arise due to the cell border crossing. Such delays are very likely to cause errors when taking into consideration the high bit-rate and the real-time nature of the information exchanged. Re-routing the handover signalling from the fixed network to the air interface will substantially reduce the overall time for performing handover in such critical applications. However, data applications are expected to occupy a multitude of slots depending on the application and its bandwidth requirements. Hence it is recommended that an investigation of the performance of the new handover strategy using PRMA as a medium access protocol with flexible slot assignment mechanism be performed. The use of priority management in the slot assignment process will be crucial to ensure that all applications are provided with the necessary resources within the specified time constraints. It is also important to notice that with real-time image transmission, a visual contact is always maintained even when there is no new data to transmit.

Hence, any transmission algorithms used for this purpose should be able to accommodate the statistical multiplexing characteristic of the PRMA technique.

- To achieve global communications a roaming facility is required to allow the mobile terminals to access and maintain a communication not only between the different layers of a single network (macrocells and microcells), but between networks of different standards (terrestrial and satellite networks). Currently, in GSM, roaming is operational where a GSM mobile terminal can operate in any GSM network. However, handing over active communication between two separate GSM networks is not implemented, despite the fact that it is a functional requirement of the GSM standard. It is therefore recommended that a further investigation is carried out that considers the decentralisation of the handover and registration procedures, particularly transferring the control of these functions from the fixed network to the mobile terminal. Furthermore, the substantial improvement obtained by the new handover strategy on a single network level prompts the need for an evaluation of this strategy between networks of the same standard, and those of different standards.

References

[1] C. J. Hughes, and M. S. Appleyby, :"Definition of a cellular mobile radio system." , IEE Proc., Vol.132, Part F, No.5, pp.416, Aug. 1985.

[2] R. L. Brewster, "Data communications and networks 3.", IEE Telecommunications Series 31, 1994.

[3] S. M. Redl, M. K. Weber, and M. W. Oliphant "An Introduction to GSM", Artech House, London 1995.

[4] P. W. Bair, P. Jung and A. Klein, "Taking the challenge of Multiple Access for Third- Generation cellular mobile radio systems- A European view", IEEE Communications Magazine, pp.82-89, Feb. 1996.

[5] K. Pahlavan, T. H. Probert and M. E. Chase, "Trends in Local Wireless Networks.", IEEE Communications Magazine, pp.88-95, March 1995.

[6] E. Damosso and G. de Brito, "COST 231 Achievements as a support to the development of UMTS: A look into the future.", IEEE Communications Magazine, pp.90-96, Feb. 1996.

[7] J. C. S. Cheung, M. A. Beach and J. P. McGeehan, "Network planning for Third-Generation mobile radio systems.", IEEE Communications Magazine, pp.54-59, Nov. 1994.

[8] F. Leite, R. Engelman, S. Kodama, H. Mennenga and S. Towaij, "Regulatory considerations relating to IMT-2000.", IEEE Personal Communications, pp.14-19, Aug. 1997.

[9] J. DeVile, "A reservation multiple access scheme for an adaptive TDMA air interface.", 4th WINLAB workshop, Nov. 19, 1993.

[10] J. S. Dasilva, D. Ikonomou, and H. Erben, "European R&D programs on Third-Generation mobile communication systems.", IEEE Personal Communications, pp.46-52, Feb. 1997.

[11] Technical Specifications GSM 5.08: Radio Subsystem Link Control.

[12] W. T. Webb, J. E. B. Williams, and R. Steele, "Microcellular teletraffic levels.", IEE Conference on Mobile and Personal Communications, 13-15 Dec. 1993, Conference Publications No. 387.

[13] R. Steele, J. Williams, D. Chandler, S. Dehghan, and A. Collard, "Teletraffic performance of GSM900/DCS1800 in street microcells.", IEEE Communications Magazine, pp.102-108, March 1995.

[14] M. Ruggieri and F. Santucci, "Design of handover procedures for mobile cellular systems through performance charts.", Wireless Personal Communications Conference, 1997, pp.51-73

[15] M. Gudmundson, "Analysis of Handover Algorithms.", Proc. Vehicular Technology Conference, St. Louis, MO, May 19-21, 1991, pp.537-542.

[16] V. Kappor, G. Edwards, and R. Sankar, "Handoff criteria for personal communication networks." , Proc. ICC '94, New Prleans, LA, May 1-5, 1994, pp.1297-1301.

[17] M. E. Anagnostou and G. C. Manos," Handover related performance of mobile communication networks.", Proc. Vehicular Technology Conference '94, Stockholm, Sweden, June 8-10, 1994, pp.111-114.

[18] K. S. MierHellstern, G. P. Pollini, D. J. Goodman, "Network protocols for the cellular packet switch.", IEEE Transactions on Communications, Feb. 1994, Vol.42, No. 2-4, Part 2, pp. 1235-1244.

[19] V. Werbus, A. Veloso, and A. Villanueva, "DECT- Cordless Functionality in New Generation Alcatel PABXs.", Electrical Communication-2nd Quarter 1993, pp.172-180.

[20] Technical Specifications (GSM 4.08): Mobile Radio Interface, Layer 3 Specifications.

[21] O. Grimlund and B. Gudmundson, "Handoff strategies in microcellular systems.", Proc. Vehicular Tech. Conf. '91, St Louis, MO, May 19-22, 1991, pp. 505-510.

[22] Y-B. Lin, S. Mohan, and A. Noerpel, "PCS Channel assignment strategies for handoff and initial access.", IEEE Personal Communications, 3rd Quarter 1994, pp. 47-56.

[23] D. J. Goodman, "Trends in cellular and cordless communications.", IEEE Communications Magazine, pp. 31-40, June 1991.

[24] W. H. W. Tuttlebee, "Cordless personal communications.", IEEE Communications Magazine, pp. 42-53, Dec. 1992.

[25] L. R. Hu and S. S. Rappaport, "Personal Communication systems using multiple hierarchical cellular overlays.", IEEE Journal on Selected Areas in Communications, Vol. 13, No. 2, Feb. 1995, pp. 406-415.

[26] S. S. Rappaport and R. L. Hu, "Microcellular communication systems with hierarchical macrocell overlays: traffic performance models and analysis." Proc. IEEE, vol. 82, no. 9, pp. 1383-1397, Sep. 1994.

[27] R. Steele and M. Nofal, "Teletraffic performance of microcellular personal communication networks.", IEE Proc.-I, Vol. 139, No. 4, August 1992, pp. 448-460.

[28] G. Colombo, C. Eynard, A. Gamba, and L. Scarpelli, " Signalling and traffic issues in personal communications.", CSELT Technical Reports, Vol. XX, No. 3, June 1992.

[29] D.J. Harasty, L. F. Chang and A. R. Noerpel, "Radio link access for a Wireless Access Communications System.", Conference Record - International Conference on Communications, 1994, Vol.2, pp.1111-1115.

[30] S. Tekinary and B. Jabbari, "A measurement based prioritization scheme for handovers in cellular and microcellular networks.", IEEE Journal Selected Areas in Communications, Oct. 1992, pp.1343-1350.

[31] T. Fujii, "Selective Handover for traffic balance in mobile communications.", Proc. IEEE/Supercomm/ICC '92, Vol. 4, 1992, pp. 212.3.1-212.3.7.

[32] D. Everitt, "Traffic capacity of Cellular Mobile Communications Systems.", Computer Networks ISDN Systems, Vol. 20, 1990, pp.447-454.

[33] Y-B. Lin, A. Noerpel, and D. A. Harasty, "Non-blocking channel assignment strategy for hand-offs.", Proc. IEEE ICUPC, 1994.

[34] A. R. Noerpel, Y. B. Lin, and H. Sherry, " PACS: Personal Access Communications System - A tutorial.", IEEE Personal Communications, June 1996, pp. 32-42.

[35] S. Chia, "Cellular structure and handover algorithms for a Third Generation mobile system.", BT Technology Journal, Jan 1993, Vol.11, No.1, pp. 111-116.

[36] G. P. Pollini, "Trends in Handover Design", IEEE Communications Magazine, March 1996, pp.82-90.

[37] S. S. Rappaport, "The multiple-call handoff problem in high capacity cellular communications systems.", IEEE Transactions on Vehicular Technology, Vol. 40, No. 3, Aug. 1991, pp. 546-557.

[38] C. Purzynski and S. S. Rappaport, "Multiple-call handoff problem with queued handoffs and mixed platform types.", IEEE Proc.-Commun., Vol. 142, No. 1, Feb. 1995, pp. 31-39.

[39] J. Sarnecki, C. Vendorai, A. Javed, P. O'kelly, and K. Dick, "Microcell design principles.", IEEE Communications Magazine, April 1993, pp. 76-82.

[40] J. Naslund et al., "An Evolution of GSM.", Proc. Vehicular Technology Conference '94, Stockholm, Sweden, June 8-10, 1994, pp. 348-352.

[41] K. Inanov and G. Spring, "Mobile speed sensitive handover in a mixed cell environment." Proc. Vehicular Tech. Conf. '95, Chicago, IL, July 25-28, 1995, pp. 892-896.

[42] S. Rappaprt, "Blocking, handoff and traffic performance for cellular communication systems with mixed platforms." , IEE Proceedings-I, Vol.140, No. 5, pp.389-401, 1993.

[43] M. Benveniste, "Cell selection in two-tier microcellular/macrocellular systems.", Proc. Globecom '95, Singapore, Nov. 1995.

[44] H. Eriksson et al, "Multiple access options for cellular based personal communications.", Proc. Vehicular Technology Conference, Secaucus, NJ, May 18-20, 1993, pp. 957-962.

[45] E. Del Re, "Satellite mobile system integrated with the terrestrial cellular network.", Conference Record - International Conference on Communications, 1989, Vol.1, pp.223-227.

[46] E. Del Re, F. D. Priscoli, R. Menolascino, I. Mistretta and F. Settimo, "Issues for he integration of satellite and cellular networks for mobile communications," in workshop Adv. Network and Tech. Concepts for Mobile, Micro and Personal communications, JPL, Pascadena, CA, May 30-31, 1991.

[47] F. D. Prescoli, "Assessment of a public mobile satellite system compatible with he GSM cellular network.", Int. Journal of Satellite Communications, Vol. 12, 2-24, 1994, pp. 13-24.

[48] E. Del Re and P. Iannucci, "The GSM procedures in an integrated cellular/satellite system." IEEE J. Selected Areas in Communications, Vol. 13, No. 2, Feb. 1995, pp. 421-430.

[49] F. Dosiere, T. Zein, G. Maral and J. Boutes, " A model for the handover traffic in Low Earth-Orbiting (LEO) satellite networks for personal communications." IEEE Global Telecommunications Conference, 1993, Vol.1, pp. 574-578.

[50] E. Del Re, R. Fantacci, G. Giambene, "Efficient dynamic channel allocation techniques with handover queuing for mobile satellite networks.", IEEE J. Selected Areas in Communications, Vol. 13, No. 2, Feb. 1995, pp. 397-404.

[51] J. Lodge, "Mobile satellite communication systems: Toward global personal communications.", IEEE Communications Magazine, Vol. 29, Jan. 1991, pp. 24-30.

[52] F. Ananasso, G. Rondinelli, P. Palmucci, and B. Pavesi, "Small satellite applications: a new perspective in satellite communications." AIAA 14th Int. Comm. Satellite Systems Conference, Washington DC, March 22-26, 1992, pp. 911-915.

[53] D. Grillo and G. Macnamee, "European perspective on Third Generation communication systems.", Proc. 40th IEEE Vehicular Technology Conference 1990, pp. 135-140.

[54] D. V. Bozward and R. L. Brewster, "Techniques to improve the capacity of a DS-CDMA system.", IEE Colloquium on Spread Spectrum techniques for radio communication systems, London, 27th April, 1993.

[55] W. C. Y. Lee, "Overview of cellular CDMA", IEEE Transactions on Vehicular Technology, Vol.40, No. 2, pp. 291-302, May 1991.

[56] G. R. Cooper and R. W. Nettleton, "A spread spectrum technique for high capacity mobile communications.", IEEE Transactions on Vehicular Technology Vol. VT-27, pp. 264-275, Nov. 1978.

[57] M. Frullone, G. Riva, P. Grazioso and C. Carciofi, "Reservation policies for packet voice protocols in cellular environments.", Mobile and Personal Comm. Conf., 13-15 Dec. 1993, pp. 142-146.

[58] J. Gruber and L. Strawcynski, "Subjective effects of variable delay and speech clipping in dynamically managed voice systems.", IEEE Trans. on Commun., Vol. COM-33, No. 8, pp. 801-808, Aug. 1985.

[59] M. Zorzi and R. Rao, "Capture and re-transmission control in mobile radio.", IEEE J. Selected Areas in Comm. Vol. 12, No. 8, Oct. 1994, pp. 1289-1298.

[60] D. J. Goodman and S. X. Wei, "Efficiency of packet reservation multiple access.", IEEE Trans. on Vehicular Tech., Vol. 40, No. 1, Feb. 1991, pp. 170-176.

[61] M. Frullone, G. Falciasecca, P. Grazioso, G. Riva and A. Serra, "On the performance of packet reservation multiple access with fixed and dynamic channel allocation.", IEEE Trans. On Vehicular Tech., Vol. 42, No. 1, Feb. 1993, pp. 78-85.

[62] F. Li and L. Merakos, "Voice/Data channel access integration in TDMA digital cellular networks.", IEEE Trans. on Vehicular Tech., Vol. 43, No. 4, Nov. 1994, pp. 986-996.

[63] D. J. Goodman, " Wireless Personal Communications Systems.", Addison-Wesley Wireless Communications Series, 1997.

[64] M. Toloo and S. Abu-Hakima, "All successful channel occupations in PRMA-DCA mobile wireless PCS. " 1997 International Symposium on Personal, Indoor and Mobile Radio Communications, Sep. 1997, Vol. 2, pp.411-414.

[65] J. Brecht,, L. Hanzo, and M. Del Buono, " Multi-frame packet reservation multiple access for variable-rate multimedia users." , 1997 International

Symposium on Personal, Indoor and Mobile Radio Communications, Sep. 1997, Vol. 2, pp.430-434.

[66] A. E. Brand and A. H. Aghvami, " Multi-dimensional PRMA (MD-PRMA) - A versatile medium access strategy for the UMTS mobile to base station channel.", 1997 International Symposium on Personal, Indoor and Mobile Radio Communications, Sep. 1997, Vol. 2, pp.524-528.

[67] J. Dunlop, "Packet access mechanisms for cellular radio.", Electronics and Comm. Eng. Journal, June 1993, pp. 173-179.

[68] P. Narasimhan and R. Yates, "A new protocol for the integration of voice and data over PRMA.", IEEE Journal on Selected Areas in Communications, May 1996, Vol.14, No. 4, pp. 623-631.

[69] N. Mitrou, G. Lyberopoulos, and A. Panagopoulou, "Voice and data integration in the air-interface of a microcellular mobile communication system.", IEEE Trans. on Vehicular Tech., Vol. 42, No. 1, Feb. 1993, pp. 1-13.

[70] W-C. Wong, C-E. Sundberg, and N. Seshardi, "Shared time division duplexing: An approach to low-delay high-quality wireless digital speech communications.", IEEE Trans. on Vehicular Tech., Vol. 43, No. 4, Nov. 1994, pp. 934-944.

[71] R. Tafazoli, M. Richharia, and B. Evans, "Multiple access schemes for future land mobile communication systems.", IEE Fifth Int. Conf. on Mobile Radio and Personal Communications, University of Warwick, UK. 11-14 Dec. 1989.

[72] P. Taaghol, R. Tafazoli, and B. Evans, "Performance optimisation of packet reservation multiple access through novel permission probability strategies.", IEE Colloquium (Digest), 1996, No. 056, pp. 5.

[73] J. Silva, B. Barani, and B. Arroyo-Fernandez," European mobile communications on the move.", IEEE Comm. Mag., Feb. 1996, pp. 60-69.

[74] J. Cosmas, B. Evans, C, Evci, W. Herzig, H. Persson, J. Petttifor, P. Polese, R. Rheinschmitt, and A. Samukic, "Overview of the mobile communications programme of RACE II.", Electronics and Comm. Eng. Journal, Aug. 1995, pp. 155-167.

[75] J. Dunlop, P. Cosmini, and D. Robertson, "Optimization of packet access mechanisms for advanced time division multiple access (ATDMA).", IEE Mobile and Personal Comm. Conf., 13-15 Dec. 1993, pp.237-242.

[76] J. Dunlop, P. Cosmini, D. Maclean, D. Robertson, and D. Aitken, "A reservation based access mechanism for 3rd generation cellular systems.", Electronics and Comm. Eng. Journal, June 1993, pp. 180-186.

[77] J. Kahwa, and N. Georganas, "A hybrid channel assignment scheme in large scale cellular- structured mobile communication systems.", IEEE Trans. on Comm., Vol. COM.26, 1978, pp. 432-438.

[78] P. Johri, "An insight into dynamic channel assignment in cellular mobile communication systems. "Euro. J. Operational Research, Vol. 74, 1994, pp. 70-77.

[79] M. Zhang and T-S. Yum, "The non-uniform compact pattern allocation algorithm for cellular mobile systems.", IEEE Trans. on Vehicular Tech., Vol. VT-40, 1991, PP. 387-391.

[80] M. Zhang, "Comparisons of channel assignment strategies in cellular mobile telephone systems.", IEEE Trans. on Vehicular Tech., Vol. VT-38, 1989, pp. 211-215.

[81] S. Tekinary, and B. Jabbari, "Handover and channel assignment in mobile cellular networks.", IEEE Comm. Mag., Vol. 29, 1991.

[82] S. Jordan, and A. Khan, " Performance bound on dynamic channel allocation in cellular systems: equal load.", IEEE Transactions on Vehicular Technology, May 1994, Vol. 43, No. 2, pp. 333-334.

[83] R. Beck, and H. Panzer, "Strategies for handover and dynamic channel allocation in micro-cellular mobile radio telephone systems.", IEEE Trans. on Vehicular Tech., Vol. 1, 1989, pp. 178-185.

[84] D. Everitt and N. W. MacFayden, " Analysis of multicellular mobile radio telephone systems with loss." BT Tech. J. Vol. 2, 1983, pp. 37-45.

[85] I. ChihLin, and C. PiHui, "Local packing- Distributed dynamic channel allocation at cellular base station." IEEE Global Telecommunications Conference, 1993, Vol.1, pp. 293-301.

[86] M. Serizawa, and D. Goodman," Instability and deadlock of distributed dynamic channel allocation." Proc. 43rd, IEEE VTC, 1993, pp. 528-531.

[87] J. Vucetic, "A hardware implementation of channel allocation algorithm based on a space-bandwidth model of a cellular network.", IEEE Trans. on Vehicular Tech., Vol. 42, 1993, pp. 444-455.

[88] F. Halsall, "Data Communications, Computer Networks and Open Systems.", Addison-Wesley Publishing Company, 1996.

[89] Technical Specifications (GSM 4.07): Mobile Radio Interface, Signalling Layer 3, General Aspects.

[90] Technical Specifications (GSM 4.88): Mobile Radio Interface Layer 3, Call Offering Supplementary Services Specifications.

[91] 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, May 1995, pp. 229235.

[92] M. Serizawa and D. J. Goodman, "Traffic sharing scheme for distributed dynamic channel allocation.", IEE Conference on Mobile and Personal Communications, 13-15 Dec. 1993, Publications No. 387.

[93] K. Madani and H. Aghvami, "DCCA: A distributed control channel allocation scheme for microcellular radio communication networks.", IEE Conference on Mobile and Personal Communications, 13-15 Dec. 1993, Publications No.387.

[94] Technical Specifications (GSM 3.09): Handover Procedure.

[95] Technical Specifications (GSM 4.13): Performance Requirements on the Mobile Radio Interface.

[96] Technical Specifications (GSM 5.08): Radio Subsystem Link Control.

[97] Technical Specifications (GSM 4.05): MS-BSS Data Link Layer, General Aspects.

[98] Technical Specifications (GSM 4.06): MS-BSS Interface Data Link Layer

[99] J. Dunlop and D. G. Smith, "Telecommunication Engineering.", Chapman & Hall, London, 1994.

[100] W. C. Y. Lee, "Smaller cells for greater performance.", IEEE Communications Magazine, Nov. 1991, pp.19-23.

[101] J. Dunlop, J. Irvine, D. Robertson, and P. Cosmini, "Performance of a statistically multiplexed access mechanism for a TDMA radio interface.", IEEE Personal Communications, June 1995, pp. 56-64.

[102] T. Benkner and K. David, "Autonomous slot assignment schemes for PRMA++ Third Generation TDMA systems.", Proc. 1995 IEEE 3 Symp. Commun. Vehicle. Technol. Benelux, 1995, pp.13-19.

[103] D. J. Goodman and S. X. Wei, "Efficiency of packet reservation multiple access.", IEEE Transactions on Vehicular Technology, Vol. 40, No. 1, February 1991, pp.170-176.

[104] B. Senadji, "A new handover decision algorithm for cellular and microcellular systems.", 1996 Fourth International Symposium on Signal Processing and its Applications, Vol. 1, pp. 228-231.

[105] R. D. Carsello, R. Meidan, S. Allpress and F. O'Brien, "IMT-2000 Standards: Radio aspects.", IEEE Personal Communications, Aug. 1997, pp.30-40.

[106] K. Buchanan, R. Fudge, and D. McFarlane, "IMT-2000: Service provider's perspective.", IEEE Personal Communications, Aug. 1997, pp. 8-13.

[107] J. Moura, R. Jasinschi, and H. Shiojiri, "Video over wireless.", IEEE Personal Communications, Feb. 1996, Vol. 3, No. 1, pp. 44-45.

APPENDICES

```
/******************************************************************************/
/*                    GSM     FCA for HO & Call initiation.                  */
/******************************************************************************/

#include <iostream.h>
#include <fstream.h>
#include <stdlib.h>
#include <iomanip.h>
#include <time.h>
#include <math.h>
#include "my_lib.h"

float greatest=0.0;
float longest=0.0;
unsigned int fail_count=0;
unsigned long channel_failed_HO=0;
unsigned long delay_failed_HO=0;
int terminal_no;
int next_channel=0;
int channel_per_cell=16;
int total_channels=224;
int no_of_cells=14;
int canddate=2;                 //  candidate cells for Handoff //
int new_cell[2];                // new_cell[canddate] //
int channel[14][16];            // channel[no_of_cells][channel_per_cell] //
int cell;
int hello=0;
int set_limit=(total_channels/2)*8;

unsigned long new_call_block=0;
unsigned special_block=0;
unsigned long new_call_start=0;
unsigned long HO_terminated_call=0;
unsigned long HO_perf=0;
unsigned int call_finished=0;
unsigned int ongoing_call=0;

float slot_period=577.0;
float call_period[8][224];          // call_period[8][total_channels] //
unsigned long  time1[8][224];       // time1[8][total_channels]       //
unsigned long next_call[8][224];    // next_call[8][total_channels]   //
unsigned long next_HO[8][224];      // next_HO[8][total_channels]     //
unsigned long global_timer=0;
char outcome;
char burst[2][8][224;

char file_name_1[]="OUT_1.TXT";
ofstream out_file_1=file_name_1;
char file_name_2[]="OUT_2.TXT";
ofstream out_file_2=file_name_2;
char file_name_3[]="OUT_3.TXT";
ofstream out_file_3=file_name_3;
char file_name_4[]="OUT_4.TXT";
ofstream out_file_4=file_name_4;
```

```
char file_name_5[]="OUT_5.TXT";
ofstream out_file_5=file_name_5;


class Pair
    {
    public:
    int ch_1;
    int slt_1;
    int ch_2;
    int slt_2;
    float figure;
    };
Pair set[896];                          //  set[set_limit]  //


char Handoff(unsigned long global_timer,int slot_1,int channel_1,unsigned int ongoing_call,int cell,char
burst[2][8][224],int new_cell[2])
{

fail_count=0;
char result;                        // to indicate the result of HO //
int channel_2=0;
int slot_2=0;

float Q_Timer_Delay=0;
float Q_Timer_Expiry=2000000;           // Queuing Timer at Target  //
float T8_Timer_Delay=0;                 // T8 Timer (Rec.Value)     //
float T8_Timer_Expiry=650000;           // Base Station (Rec.Value) //
float PLMN_delay=1000000*ongoing_call*3.9e-5;

float delay=577                         // Measurement Report (ISDN) //
unsigned long old_timer=global_timer;

if (cell==5)
    { new_cell[0]=10;
    new_cell[1]=9; }

if (cell==6)
    {
    new_cell[0]=9;
    new_cell[1]=10;
    }

if (cell==9)
    {
     new_cell[0]=6;
    new_cell[1]=5;
    }

if (cell==10)
    {
    new_cell[0]=5;
     new_cell[1]=6;
    }
                        // (0.1*random(6)) gives a value between 0-0.5 //
```

```
delay=delay+PLMN_delay;                      // HO_Required (ISDN) //
Q_Timer_Delay=PLMN_delay;
delay=delay+PLMN_delay;                      // HO_Request (ISDN) //


/*********************** Looking for an idle channel *********************************/
for (int i=0; i<canddate;i++)
    {
    for (int k=0; k<channel_per_cell; k++)
        {
        for (int p=0; p<=7; p++)
            {
            int tempo=channel[new_cell[i]][k];
            if (burst[0][p][tempo]=='i')
                {
                channel_2=tempo;
                slot_2=p;
                goto step1;
                }
            }
        }
    }

/***************************************************************************/
burst[0][slot_1][channel_1]='i';    // reset a failed HO & change //
burst[1][slot_1][channel_1]='i';    // the old channel to idle    //

call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
return result='F';

step1:

if (Q_Timer_Delay>=Q_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';    // reset a failed HO & change //
    burst[1][slot_1][channel_1]='i';    // the old channel to idle    //

    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;
    return result='F';
    }

delay=delay+PLMN_delay;        // HO_Request_Ack (ISDN) //
T8_Timer_Delay=PLMN_delay;
delay=delay+PLMN_delay;                 // HO_Command  (ISDN)    //

/******************** T8 timer starts here ****************/
delay=delay+577;              // HO_Command  (downlink) //
T8_Timer_Delay=T8_Timer_Delay+577;

/*************** Check for T8 expiry *********************/
if (T8_Timer_Delay>=T8_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';    // reset a failed handoff    //
    burst[1][slot_1][channel_1]='i';    // and change channel to idle //

    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;
```

```
    return result='F';
    }
/******************************************************************************/
delay=delay+577;          // HO_Ref_Number (uplink) //
T8_Timer_Delay=T8_Timer_Delay+577;

/********** Check for T8 expiry ********************/
if (T8_Timer_Delay>=T8_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';  // reset a failed handoff    //
    burst[1][slot_1][channel_1]='i';  // and change channel to idle //

    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;
    return result='F';
    }
 /**************************************************/
delay=delay+577;          // HO_Detect (downlink) //
T8_Timer_Delay=T8_Timer_Delay+577;

/********** Check for T8 expiry ********************/
if (T8_Timer_Delay>=T8_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';     // reset a failed handoff    //
    burst[1][slot_1][channel_1]='i';     // and change channel to idle //

    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;

    return result='F';
    }
/******************************************************************************/
delay=delay+577;          // HO_Complete (uplink) //
T8_Timer_Delay=T8_Timer_Delay+577;

/********** Check for T8 expiry ********************/
if (T8_Timer_Delay>=T8_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';     // reset a failed handoff    //
    burst[1][slot_1][channel_1]='i';     // and change channel to idle //

    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;
    return result='F';
    }
/******************************************************************************/

delay=delay+PLMN_delay;                          // HO_Complete (ISDN) //
T8_Timer_Delay=T8_Timer_Delay+PLMN_delay;

/********** Check for T8 expiry ********************/
if (T8_Timer_Delay>=T8_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';     // reset a failed handoff    //
    burst[1][slot_1][channel_1]='i';     // and change channel to idle //

    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;
```

```
    return result='F';
    }
/*********************************************************************************/
delay=delay+PLMN_delay;      // Clear_Command (ISDN) //
delay=delay+PLMN_delay;      // Clear_Complete (ISDN) //

burst[0][slot_1][channel_1]='T';
burst[1][slot_1][channel_1]='T';

burst[0][slot_2][channel_2]='H';
burst[1][slot_2][channel_2]='H';


/*************** finding an available set to allocate the Handover channel pair *************/

for(i=0; i<set_limit; i++)
    {
    if (set[i].figure==0.0)
        {
        set[i].figure=old_timer+delay;
        set[i].ch_1=channel_1;
        set[i].slt_1=slot_1;
        set[i].ch_2=channel_2;
        set[i].slt_2=slot_2;
        goto step3;
        }
    }

step3:

result='S';
return result;

}
/*********************************** MAIN ***********************************/

void main()
{

time_t t;
srand(unsigned (time(&t)));

out_file_1 << "Handover Dealy in Ascending order " << endl;
out_file_3 << "Number of Ongoing calls " << endl;
out_file_4 << "Poisson call rate periods " << endl;

for (int x=0; x<set_limit; x++)
    {
    set[x].ch_1=0;
    set[x].slt_1=0;
    set[x].ch_2=0;
    set[x].slt_2=0;
    set[x].figure=0;
    }

for (int y=0; y<=7; y++)
    {
    for (int k=0; k<total_channels; k++)
        {
```

```
        next_HO[y][k]=0;
        next_call[y][k]=0;
        call_period[y][k]=0;
        }
    }

/*************************** CHANNEL ASSIGNEMENT ***********************/
int number=0;

for (int i=0; i<no_of_cells ; i++)
    {
    for (int j=0; j<channel_per_cell; j++)
        {
        channel[i][j]=number;
        number=number+1;
        }
    }

/******************** FUNCTION TO MAKE IDLE CHANNELS *******************/

for (int v=0; v<=7 ;v++)
    {
    for (int y=0; y<total_channels; y++)
        {
        burst[0][v][y]='i';
        burst[1][v][y]='i';
        }
    }


/*********** FUNCTION TO DEFINE THE NO. OF TERMINALS ********************/
int dan=0;

cout << "1- 100%  capacity 1792 terminals" << endl;
cout << "2- 50%   capacity 896  terminals" << endl;
cout << "3- 25%   capacity 448  terminals" << endl;
cout << "4- 12.5% capacity 224  terminals" << endl;
cout << "5- 6.25% capacity 112  terminals" << endl;
cout << endl << endl;
cout << "Enter your choice (1-5): ";
cin >> dan;

switch(dan)
{
case (1):
    terminal_no=1;
    break;
    case (2):
    terminal_no=2;
    break;
    case (3):
    terminal_no=4;
    break;
    case (4):
    terminal_no=8;
    break;
    case (5):
    terminal_no=16;
    break;
```

```
    }

/*********************** SIMULATION STARTS HERE ****************************/

for (long u=0; u<65000; u++)
{
for (int x=0; x<=7 ;x++)
    {
    global_timer=global_timer+slot_period;
    next_channel=0;
    for (int cell=0; cell<no_of_cells; cell++)     /**** cell loop ****/
        {
        for (int z=0; z<channel_per_cell; z++)     /**** channel loop ****/
            {


/******** FUNCTION TO GENERATE HAND-OFF REQUESTS AT Poisson RATE *******/
/*                                                                      */
/* Perform inter-BSC Hanodover in the cells 5,6,9 & 10 only             */
/************************************************************************/


if ((cell==5) || (cell==6) || (cell==9) || (cell==10))
    {
    if((next_HO[x][channel[cell][z]]==0) && (burst[0][x][channel[cell][z]]=='D'))
        {
        next_HO[x][channel[cell][z]]=global_timer+(1000000*Poisson_10());
        goto step_x;
        }

    if ((global_timer>=next_HO[x][channel[cell][z]]) && (burst[0][x][channel[cell][z]]=='D'))
        {
        outcome=Handoff(global_timer,x,channel[cell][z],ongoing_call,cell,burst,new_cell);
        if (outcome=='S')  HO_perf=HO_perf+1;
        if (outcome=='F')
            {
            HO_terminated_call=HO_terminated_call+1;
            ongoing_call=ongoing_call-1;
            }
        if ((outcome=='F') && (fail_count==0)) channel_failed_HO=channel_failed_HO+1;
        if ((outcome=='F') && (fail_count >0)) delay_failed_HO=delay_failed_HO+1;
        next_HO[x][channel[cell][z]]=0;
        goto step2;
        }
    }

                    //\/\/\/\/\/\/\/ end of Handover function  \/\/\/\/\/\/\/\/\/\/\/

/************** FUNCTION TO GENERATE CALLS AT Poisson RATE ***************/


if (channel[cell][z]==next_channel)
    {
    if (next_call[x][channel[cell][z]]==0)
        {
        next_call[x][channel[cell][z]]=global_timer+(144000000);
        out_file_4 << next_call[x][channel[cell][z]]-global_timer << endl;
        next_channel=channel[cell][z]+terminal_no;
        goto step_x;
        }
```

```
    if (global_timer>=next_call[x][channel[cell][z]])
        {
        for (int j=0; j<channel_per_cell; j++)
            {
            for (int k=0; k<=7; k++)
                {
                if (burst[0][k][channel[cell][j]]=='i')
                    {
                    burst[0][k][channel[cell][j]]='D';
                    burst[1][k][channel[cell][j]]='D';
                    float period=1000000*Exponential_1();
                    call_period[k][channel[cell][j]]=period;
                    time1[k][channel[cell][j]]=global_timer;
                    next_call[x][channel[cell][z]]=0;
                    new_call_start=new_call_start+1;
                    ongoing_call=ongoing_call+1;
                    if (ongoing_call>greatest)
                        {
                        greatest=ongoing_call;
                        }
                    next_channel=channel[cell][z]+terminal_no;
                    goto step_x;
                    }
                }
            }
            new_call_block=new_call_block+1;

            if ((cell==5) || (cell==6) || (cell==9) || (cell==10))
                {
                special_block=special_block+1;
                }

            next_call[x][channel[cell][z]]=0;
            next_channel=channel[cell][z]+terminal_no;
            goto step_x;
        }

    }
                //\/\/\/\/\/\/\/\/\/\ end of next_call Function /\/\/\/\/\/\/\/\/\/\\

step_x:


if ((burst[0][x][channel[cell][z]]=='D') && (global_timer-
time1[x][channel[cell][z]])>=call_period[x][channel[cell][z]])
    {
    burst[0][x][channel[cell][z]]='i';
    burst[1][x][channel[cell][z]]='i';
    call_period[x][channel[cell][z]]=0;
    call_finished=call_finished+1;
    ongoing_call=ongoing_call-1;
    goto step2;
    }


if ((burst[0][x][channel[cell][z]]=='T') && (global_timer-
time1[x][channel[cell][z]])>=call_period[x][channel[cell][z]])
```

```
    {
    for (int k=0; k<set_limit; k++)
        {
        if ((set[k].slt_1==x) && (set[k].ch_1==channel[cell][z]))
            {
            burst[0][set[k].slt_2][set[k].ch_2]='i';
            burst[1][set[k].slt_2][set[k].ch_2]='i';
            burst[0][set[k].slt_1][set[k].ch_1]='i';
            burst[1][set[k].slt_1][set[k].ch_1]='i';
            time1[set[k].slt_1][set[k].ch_1]=0;
            call_period[set[k].slt_1][set[k].ch_1]=0;
            set[k].figure=0;
            set[k].ch_1=0;
            set[k].ch_2=0;
            set[k].slt_1=0;
            set[k].slt_2=0;
            call_finished=call_finished+1;
            ongoing_call=ongoing_call-1;
            goto step2;
            }
        }
}


if (burst[0][x][channel[cell][z]]=='H')
    {
    for (int k=0; k<set_limit; k++)
        {
        if ((set[k].slt_2==x) && (set[k].ch_2==channel[cell][z]))
            {
            if (global_timer>=set[k].figure)
                {
                burst[0][set[k].slt_2][set[k].ch_2]='D';
                burst[1][set[k].slt_2][set[k].ch_2]='D';
                burst[0][set[k].slt_1][set[k].ch_1]='i';
                burst[1][set[k].slt_1][set[k].ch_1]='i';
                time1[set[k].slt_2][set[k].ch_2]=time1[set[k].slt_1][set[k].ch_1];
                time1[set[k].slt_1][set[k].ch_1]=0;
                call_period[set[k].slt_2][set[k].ch_2]=call_period[set[k].slt_1][set[k].ch_1];
                 call_period[set[k].slt_1][set[k].ch_1]=0;
                set[k].figure=0;
                set[k].ch_1=0;
                set[k].ch_2=0;
                set[k].slt_1=0;
                set[k].slt_2=0;

                goto step2;
            }
            goto step2;
        }
    }
}

step2:

            }            /**** z (channel) loop ****/
        }                /**** cell counter loop ****/
    }                    /**** x (slot) loop     *****/
```

```
    }                       /**** U loop        ******/

cout << endl << endl;

cout << "Max ongoing_call= " << greatest << endl;
cout << "Max HO_delay= " << longest << endl;
cout << "new_call_start= " << new_call_start << endl;
cout << "call_finished= " << call_finished << endl;
cout << "new_call_block= " << new_call_block << endl;
cout << "special_block= " << special_block << endl;
cout << "HO_terminated_call= " << HO_terminated_call << endl;
cout << "channel_failed_HO= " << channel_failed_HO << endl;
cout << "delay_failed_HO= " << delay_failed_HO << endl;
cout << "HO_performed= " << HO_perf << endl;
cout << "global_timer= " << global_timer << endl;


}    /************ main function ************/
```

```
/******************************************************************************/
/*                          DCA for HO & Call initiation                       */
/******************************************************************************/

#include <iostream.h>
#include <fstream.h>
#include <stdlib.h>
#include <iomanip.h>
#include <time.h>
#include <math.h>
#include "my_lib.h"


float greatest=0.0;
float longest=0.0;
unsigned int fail_count=0;
unsigned long channel_failed_HO=0;
unsigned long delay_failed_HO=0;
int zz=0;
int xx=0;

int terminal_no;
int channel_per_cell=16;
int total_channels=224;
int location[224];
int no_of_cells=14;
int canddate=2;              // candidate cells for Handoff //
int new_cell[2];             // new_cell[canddate] //
int channel[14][16];         // channel[no_of_cells][channel_per_cell] //
int cell;
int hello=0;
int set_limit=(total_channels/2)*8;
unsigned long rino=0;


unsigned long    new_call_block=0;
unsigned         special_block=0;
unsigned long    new_call_start=0;
unsigned long    HO_terminated_call=0;
unsigned long    HO_perf=0;
unsigned int     call_finished=0;
unsigned int     ongoing_call=0;

float            slot_period=577.0;
float            call_period[8][224];          // call_period[8][total_channels] //
unsigned long    time1[8][224];          // time1[8][total_channels]     //
unsigned long    next_call[8][14][16];
unsigned long    next_HO[8][224];        // next_HO[8][total_channels]    //
unsigned long    global_timer=0;
char outcome;
char burst[2][8][224];       /*************************************/
                             /*     burst[0-1][0-7][total_channels]       */
                             /*************************************/

char file_name_1[]="OUT_1.TXT";
ofstream out_file_1=file_name_1;
```

```
char file_name_2[]="OUT_2.TXT";
ofstream out_file_2=file_name_2;

char file_name_3[]="OUT_3.TXT";
ofstream out_file_3=file_name_3;

char file_name_4[]="OUT_4.TXT";
ofstream out_file_4=file_name_4;


char file_name_5[]="OUT_5.TXT";
ofstream out_file_5=file_name_5;


char file_name_6[]="OUT_6.TXT";
ofstream out_file_6=file_name_6;

class Pair
   {
    public:
     int ch_1;
     int slt_1;
     int ch_2;
     int slt_2;
     float figure;
   };
Pair set[896];          //  set[set_limit]  //


/****************************************************************************/
char Handoff(unsigned long global_timer,int slot_1,int channel_1,unsigned int ongoing_call,int cell,char
burst[2][8][224],int new_cell[2])
{

fail_count=0;
char    result;                          // to indicate the result of HO //
int     channel_2=0;
int     slot_2=0;

float   Q_Timer_Delay=0;
float    Q_Timer_Expiry=2000000;              // Queuing Timer at Target  //
float   T8_Timer_Delay=0;                     // T8 Timer (Rec.Value)     //
float   T8_Timer_Expiry=650000;               // Base Station (Rec.Value) //

float PLMN_delay=1000000*ongoing_call*3.9e-5;

float delay=577;                              // Measurement Report (ISDN) //
unsigned long old_timer=global_timer;

if (cell==5)
    {
    new_cell[0]=10;
    new_cell[1]=9;
    }

if (cell==6)
    {
    new_cell[0]=9;
    new_cell[1]=10;
```

```
    }

if (cell==9)
    {
    new_cell[0]=6;
    new_cell[1]=5;
    }

if (cell==10)
    {
    new_cell[0]=5;
    new_cell[1]=6;
    }

            // (0.1*random(6)) gives a value between 0-0.5 //

delay=delay+PLMN_delay;                    // HO_Required  //
Q_Timer_Delay=PLMN_delay;
delay=delay+PLMN_delay;                    // HO_Request  //

/*********************** Looking for an idle channel *******************************/
for (int i=0; i<canddate;i++)
    {
    for (int jj=0; jj<total_channels; jj++)
        {
        if (location[jj]==new_cell[i])
            {
            for (int xx=0; xx<8; xx++)
                {
                if (burst[0][xx][jj]=='i')
                    {
                    channel_2=jj;
                    slot_2=xx;
                    goto step_1;
                    }
                }
            }
        }
    }

    for (zz=0; zz<total_channels; zz++)
        {
        for (xx=0; xx<8; xx++)
            {
            if (burst[0][xx][zz]=='i')
                {
                channel_2=zz;
                slot_2=xx;
                location[zz]=cell;
                goto step_1;
                }
            }
        }

/*****************************************************************************/

burst[0][slot_1][channel_1]='i';    // reset a failed HO & change //
burst[1][slot_1][channel_1]='i';    // the old channel to idle    //
call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
```

```
return result='F';

step_1:

if (Q_Timer_Delay>=Q_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';   // reset a failed HO & change //
    burst[1][slot_1][channel_1]='i';   // the old channel to idle   //
    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;
    return result='F';
    }

delay=delay+PLMN_delay;        // HO_Request_Ack (ISDN) //
T8_Timer_Delay=PLMN_delay;
delay=delay+PLMN_delay;                // HO_Command  (ISDN)   //

/*********** T8 timer starts here *******************/
delay=delay+577;              // HO_Command  (downlink) //
T8_Timer_Delay=T8_Timer_Delay+577;

/********** Check for T8 expiry ********************/
if (T8_Timer_Delay>=T8_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';   // reset a failed handoff    //
    burst[1][slot_1][channel_1]='i';   // and change channel to idle //

    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;
    return result='F';
    }
/*****************************************************************************/
delay=delay+577;          // HO_Ref_Number (uplink) //
T8_Timer_Delay=T8_Timer_Delay+577;

/********** Check for T8 expiry ********************/
if (T8_Timer_Delay>=T8_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';   // reset a failed handoff    //
    burst[1][slot_1][channel_1]='i';   // and change channel to idle //

    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;

    return result='F';
    }
/*****************************************************************************/

delay=delay+577;          // HO_Detect (downlink) //
T8_Timer_Delay=T8_Timer_Delay+577;

/********** Check for T8 expiry ********************/
if (T8_Timer_Delay>=T8_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';      // reset a failed handoff    //
    burst[1][slot_1][channel_1]='i';      // and change channel to idle //
    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;
```

```
    return result='F';
    }
/*****************************************************************************/
delay=delay+577;          // HO_Complete (uplink) //
T8_Timer_Delay=T8_Timer_Delay+577;

/********** Check for T8 expiry ********************/
if (T8_Timer_Delay>=T8_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';      // reset a failed handoff    //
    burst[1][slot_1][channel_1]='i';      // and change channel to idle //

    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;

    return result='F';
    }
/*****************************************************************************/
delay=delay+PLMN_delay;                              // HO_Complete (ISDN) //
T8_Timer_Delay=T8_Timer_Delay+PLMN_delay;

/********** Check for T8 expiry ********************/
if (T8_Timer_Delay>=T8_Timer_Expiry)
    {
    burst[0][slot_1][channel_1]='i';      // reset a failed handoff    //
    burst[1][slot_1][channel_1]='i';      // and change channel to idle //

    call_period[slot_1][channel_1]=global_timer-time1[slot_1][channel_1];
    fail_count=fail_count+1;

    return result='F';
    }
/*****************************************************************************/

delay=delay+PLMN_delay;      // Clear_Command (ISDN) //
delay=delay+PLMN_delay;      // Clear_Complete (ISDN) //

burst[0][slot_1][channel_1]='T';
burst[1][slot_1][channel_1]='T';

burst[0][slot_2][channel_2]='H';
burst[1][slot_2][channel_2]='H';


/************** finding an available set to allocate the Handover channel pair ***************/

for(i=0; i<set_limit; i++)
    {
    if (set[i].figure==0.0)
        {
        set[i].figure=old_timer+delay;
        set[i].ch_1=channel_1;
        set[i].slt_1=slot_1;
        set[i].ch_2=channel_2;
        set[i].slt_2=slot_2;
        goto step3;
        }
    }
```

```
step3:

result='S';
return result;
}

/****************************** MAIN ******************************************/

void main()
{

time_t t;
srand(unsigned (time(&t)));

out_file_1 << "Handover Delay in Ascending order " << endl;
out_file_3 << "Number of Ongoing calls " << endl;
out_file_4 << "Poisson call rate periods " << endl;

for (int x=0; x<set_limit; x++)
    {
    set[x].ch_1=0;
    set[x].slt_1=0;
    set[x].ch_2=0;
    set[x].slt_2=0;
    set[x].figure=0;
    }

for (int y=0; y<=7; y++)
    {
    for (int k=0; k<total_channels; k++)
        {
        next_HO[y][k]=0;
        call_period[y][k]=0;
        }
    }

/************************* CHANNEL ASSIGNEMENT ****************************/
int number=0;

for (int i=0; i<no_of_cells ; i++)
    {
        for (int j=0; j<channel_per_cell; j++)
            {
            channel[i][j]=number;
            location[number]=i;
            number=number+1;
            for (int w=0; w<8; w++)
                {
                next_call[w][i][j]=0;
                }
            }
    }

/****************** FUNCTION TO MAKE IDLE CHANNELS *********************/
for (int v=0; v<=7 ;v++)
    {
    for (int y=0; y<total_channels; y++)
        {
```

```
        burst[0][v][y]='i';
       burst[1][v][y]='i';
       }
   }

/************** FUNCTION TO DEFINE THE NO. OF TERMINALS *******************/
int     dan=0;

cout << "1- 100%  capacity 1792 terminals" << endl;
cout << "2- 50%   capacity 896  terminals" << endl;
cout << "3- 25%   capacity 448  terminals" << endl;
cout << "4- 12.5% capacity 224  terminals" << endl;
cout << "5- 6.25% capacity 112  terminals" << endl;
cout << endl << endl;
cout << "Enter your choice (1-5): ";
cin >> dan;

switch(dan)
   {
   case (1):
   terminal_no=1;
   break;
   case (2):
   terminal_no=2;
   break;
   case (3):
   terminal_no=4;
   break;
   case (4):
   terminal_no=8;
   break;
   case (5):
   terminal_no=16;
   break;
   }
 int ram_limit=16/terminal_no;

/********************** SIMULATION START**********************************/

for (long u=0; u<65000; u++)
   {
   for (int x=0; x<=7 ;x++)
       {
       global_timer=global_timer+slot_period;
       for (int cell=0; cell<no_of_cells; cell++)    /**** cell loop ****/
           {
           int ram=0;
           for (int z=0; z<total_channels; z++)    /**** channel loop ****/
               {
               if (location[z]==cell)
                   {




/******** FUNCTION TO GENERATE HAND-OFF REQUESTS AT Poisson RATE *******/
```

```
/*
/* Perform inter-BSC Hanodover in the cells 5,6,9 & 10 only                 */
/***************************************************************************/

if ((cell==5) || (cell==6) || (cell==9) || (cell==10))
    {
    if((next_HO[x][z]==0) && (burst[0][x][z]=='D'))
        {
        next_HO[x][z]=global_timer+(1000000*Poisson_5());
        goto step_x;
        }
    if ((global_timer>=next_HO[x][z]) && (burst[0][x][z]=='D'))
        {
        outcome=Handoff(global_timer,x,z,ongoing_call,cell,burst,new_cell);

    if (outcome=='S')  HO_perf=HO_perf+1;
    if (outcome=='F')  HO_terminated_call=HO_terminated_call+1;


    if ((outcome=='F') && (fail_count==0)) channel_failed_HO=channel_failed_HO+1;
    if ((outcome=='F') && (fail_count >0)) delay_failed_HO=delay_failed_HO+1;
    next_HO[x][z]=0;
    goto step_2;
    }
}

                    //\\\\\\\\\\\ end of Handover function \\\\\\\\\\\\\\\\\/


/************ FUNCTION TO GENERATE CALLS AT Poisson RATE *****************/

if (ram<ram_limit)
    {
    if (next_call[x][cell][ram]==0)
        {
        next_call[x][cell][ram]=global_timer+(18000000);
        ram=ram+1;
        goto step_x;
        }

    if ((next_call[x][cell][ram]>0) && (global_timer<next_call[x][cell][ram]))
        {
         ram=ram+1;
        goto step_x;
        }

    if (global_timer>=next_call[x][cell][ram])
        {
        for (int j=0; j<total_channels; j++)
            {
            if (location[j]==cell)
                {
                for (int k=0; k<8; k++)
                    {
                    if (burst[0][k][j]=='i')
                        {
                        burst[0][k][j]='D';
                        burst[1][k][j]='D';
                        float period=1000000*Exponential_1();
```

```
                        call_period[k][j]=period;
                        time1[k][j]=global_timer;
                        next_call[x][cell][ram]=0;
                        new_call_start=new_call_start+1;
                        ongoing_call=ongoing_call+1;
                        ram=ram+1;
                        goto step_x;
                        }
                    }
                }
            }

    for (int jj=0; jj<total_channels; jj++)
        {
        int zimm=0;
        for (int xx=0; xx<8; xx++)
            {
            if (burst[0][xx][jj]=='i')
                {
                zimm=zimm+1;
                }
            }

        if (zimm==8)
            {
            burst[0][0][jj]='D';
            burst[1][0][jj]='D';
            float period=1000000*Exponential_1();
            call_period[0][jj]=period;
            location[jj]=cell;
            time1[0][jj]=global_timer;
            next_call[x][cell][ram]=0;
            new_call_start=new_call_start+1;
            ongoing_call=ongoing_call+1;
            ram=ram+1;
            goto step_x;
            }
        }

        new_call_block=new_call_block+1;
        if ((cell==5) || (cell==6) || (cell==9) || (cell==10))
            {
            special_block=special_block+1;
            }
        next_call[x][cell][ram]=0;
        ram=ram+1;
        goto step_x;
        }
    }

//\/\/\/\/\/\/\/\/\ end of next_call Function /\/\/\/\/\/\/\/\/\

step_x:


if ((burst[0][x][z]=='D') && (global_timer-time1[x][z])>=call_period[x][z])
    {
    burst[0][x][z]='i';
    burst[1][x][z]='i';
```

```
    call_period[x][z]=0;
    call_finished=call_finished+1;
    ongoing_call=ongoing_call-1;
    goto step_2;
    }

if ((burst[0][x][z]=='T') && (global_timer-time1[x][z])>=call_period[x][z])
    {
    for (int k=0; k<set_limit; k++)
        {
        f ((set[k].slt_1==x) && (set[k].ch_1==z))
            {
            burst[0][set[k].slt_2][set[k].ch_2]='i';
            burst[1][set[k].slt_2][set[k].ch_2]='i';
            burst[0][set[k].slt_1][set[k].ch_1]='i';
            burst[1][set[k].slt_1][set[k].ch_1]='i';
            set[k].figure=0;
            set[k].ch_1=0;
            set[k].ch_2=0;
            set[k].slt_1=0;
            set[k].slt_2=0;
            time1[set[k].slt_1][set[k].ch_1]=0;
            call_period[set[k].slt_1][set[k].ch_1]=0;
            call_finished=call_finished+1;
            ongoing_call=ongoing_call-1;

goto step_2;
            }
        }
    }


if (burst[0][x][z]=='H')
    {
    for (int k=0; k<set_limit; k++)
        {
        if ((set[k].slt_2==x) && (set[k].ch_2==z))
            {
            if (global_timer>=set[k].figure)
                {
                burst[0][set[k].slt_2][set[k].ch_2]='D';
                burst[1][set[k].slt_2][set[k].ch_2]='D';
                burst[0][set[k].slt_1][set[k].ch_1]='i';
                burst[1][set[k].slt_1][set[k].ch_1]='i';
                time1[set[k].slt_2][set[k].ch_2]=time1[set[k].slt_1][set[k].ch_1];
                time1[set[k].slt_1][set[k].ch_1]=0;
                call_period[set[k].slt_2][set[k].ch_2]=call_period[set[k].slt_1][set[k].ch_1];
                call_period[set[k].slt_1][set[k].ch_1]=0;
                set[k].figure=0;
                set[k].ch_1=0;
                set[k].ch_2=0;
                set[k].slt_1=0;
                set[k].slt_2=0;
                goto step_2;
                }
            goto step_2;
            }
        }
    }
```

```
step_2:

    }        /**** location[z] loop  ****/
}            /**** z (channel) loop  ****/

step_3:


if (ram<ram_limit)
    {
    rino=rino+1;
    if (next_call[x][cell][ram]==0)
        {
        next_call[x][cell][ram]=global_timer+(30000000);
        ram=ram+1;
        goto step_3;
        }

    if ((next_call[x][cell][ram]>0) && (global_timer<next_call[x][cell][ram]))
        {
        ram=ram+1;
        goto step_3;
        }

    if (global_timer>=next_call[x][cell][ram])
        {
        for (int j=0; j<total_channels; j++)
            {
            if (location[j]==cell)
                {
                for (int k=0; k<8; k++)
                    {
                    if (burst[0][k][j]=='i')
                        {
                        burst[0][k][j]='D';
                        burst[1][k][j]='D';
                        float period=1000000*Exponential_1();
                        call_period[k][j]=period,
                        time1[k][j]=global_timer;
                        next_call[x][cell][ram]=0;
                        new_call_start=new_call_start+1;
                        ongoing_call=ongoing_call+1;
                        ram=ram+1;
                        goto step_3;
                        }
                    }
                }
            }

    for (int jj=0; jj<total_channels; jj++)
        {
        int zimm=0;
        for (int xx=0; xx<8; xx++)
            {
            if (burst[0][xx][jj]=='i')
                {
                zimm=zimm+1;
                }
```

```
            }

            if (zimm==8)
                {
                burst[0][0][jj]='D';
                burst[1][0][jj]='D';
                float period=1000000*Exponential_1();
                call_period[0][jj]=period;
                location[jj]=cell;
                time1[0][jj]=global_timer;
                next_call[x][cell][ram]=0;
                new_call_start=new_call_start+1;
                ongoing_call=ongoing_call+1;
                ram=ram+1;
                goto step_3;
                }
            }

        new_call_block=new_call_block+1;
        if ((cell==5) || (cell==6) || (cell==9) || (cell==10))
            {
            special_block=special_block+1;
            }
        next_call[x][cell][ram]=0;
        ram=ram+1;
        goto step_3;
        }
    }

        }           /**** cell counter loop *****/
    }               /**** x (slot) loop     ******/
}                   /**** U loop            ***)))*/

for (int p=0; p<total_channels; p++)
    {
    out_file_6 << p <<"    " << location[p] << endl;
    }

cout << endl << endl;

cout << "     Max ongoing call= " << greatest << endl;
cout << "     Max HO_Delay= " << longest << endl;
cout << "     new_call_start= " << new_call_start << endl;
cout << "     call_finished= " << call_finished << endl;
cout << "     new_call_block= " << new_call_block << endl;
cout << "     special_block= " << special_block << endl;
cout << "     HO_terminated_call= " << HO_terminated_call << endl;
cout << "     channel_failed_HO= " << channel_failed_HO << endl;
cout << "     delay_failed_HO= " << delay_failed_HO << endl;
cout << "     HO_performed= " << HO_perf << endl;
cout <<"      global_timer= " << global_timer << endl;
cout << "     rino= " << rino << endl;


 }   /************ main function **********/
```

```
/*****************************************************************/
/*                          PRMA++                                */
/*                                                                */
/*  Normal PRMA operation with Handover function through PLMN     */
/*  (NO-PRIORITY).                                                */
/*  i.e. A handover terminal has to contend for slot reservation  */
/*      every time and for any channel access attempt.            */
/*                                                                */
/*   Standard Specifications                                      */
/*   ----------------------                                       */
/*  PRMA++ Transmission: Packet Size= 125 bits ==> 70 MicroSec    */
/*  Channel Rate= 1.8 Mb/s ==> 0.55 MicroSec/bit                  */
/*   Packet Dropping Threshold= 10 milliSec  (2 frames)           */
/*  slots per frame= 72                                           */
/*  Voice encoder= 13 kb/s                                        */
/*  Acceptable Packet loss Rate= 1 percent or lower               */
/*                                                                */
/*  Algorithm Specifications:                                     */
/*  ------------------------                                      */
/*  46 slots per channel (30 Info. & 16 Reservation)              */
/*                                                                */
/*      Changable specifiations:                                  */
/*      ----------------------------                              */
/*   max_terminals= start_up_limit+80                             */
/*   start_up_limit= 30*2*scaling factor                          */
/*****************************************************************/

#include <iostream.h>
#include <fstream.h>
#include <stdlib.h>
#include <iomanip.h>
#include <time.h>
#include <math.h>
#include <alloc.h>
#include <string.h>
#include "my_lib.h"

/**************** SPECIFICATIONS DATA ************************************/
int channel_per_cell=2;
int slots_per_frame=46;
int no_of_cells=7;
int total_channels=14;
int max_terminals=140          // max terminals allowed per cell  //
int start_up_limit=60;         // start_up no of terminals per cell //
unsigned int terminal_per_cell[7];
/*************************************************************************/
int hello;
int pkt_length=70;                    /***** packet_length in Micro.seconds *****/

unsigned long    average_dropped_pkt=0;
unsigned long    HO_performed_1=0;
unsigned long    HO_performed_2=0;
unsigned long    worst_dropped=0;
unsigned long    global_timer=0;
unsigned long    total_dropped_pkt=0;


unsigned int          reserve[7][2];
```

```
unsigned long far      call_period[7][140];
unsigned long far      dropped_pkt[7][140];
unsigned long far      access_delay[7][140];
unsigned long far      HO_delay[7][140];
unsigned long far      speech_pkt[7][140];
unsigned long far      silence_pkt[7][140];
unsigned long far      speech_pkt_count[7][140];
unsigned long far      silence_pkt_count[7][140];
unsigned long far       max_access_delay[7][140];

int                    ack[7][140];
unsigned char far      burst[7][140];
unsigned long far      next_HO[7][140];
unsigned int. far      user_slot[7][140];
unsigned int  far      pkt_ready[7][140];
unsigned long far       next_call[7][140];
unsigned long far      next_stage[7][140];
unsigned int  far      pkt_exp_timer[7][140];
unsigned char far      slot[644];

int set_limit=600;
int new_cell;

class Pair
    {
     public:
        unsigned int cell_1;
        unsigned int slt_1;
        unsigned int ter;
        unsigned long figure;
    };

Pair set[600];          /****** set[set_limit] *******/

char file_name_1[]="OUT_1.TXT";
ofstream out_file_1=file_name_1;

char file_name_2[]="OUT_2.TXT";
ofstream out_file_2=file_name_2;

char file_name_3[]="OUT_3.TXT";
ofstream out_file_3=file_name_3;


int contention_prob()
    {
    int k=random(40);
    return k;
    }
/**************************************************************************/
unsigned long  Handoff_1(unsigned long global_timer,int slot_1,unsigned long ongoing_call,int
cell,unsigned int terminal)
        {
        HO_performed_1++;
        unsigned long PLMN_delay=80*ongoing_call;        // 1e6 * 3.9e-5= 40  //
        unsigned long old_timer=global_timer;


   unsigned long delay=70;                 // Measurement Report //
```

```
    delay=delay+PLMN_delay;          // HO_Required       //
    delay=delay+PLMN_delay;          // HO_Request        //
    delay=delay+PLMN_delay;          // HO_Request_Ack    //
    delay=delay+PLMN_delay;          // HO_Command        //

/**************** finding an available set to allocate the Handover channel pair *************/

for(int i=0; i<set_limit; i++)
    {
    if (set[i].figure==0)
        {
        set[i].figure=old_timer+delay;
        set[i].cell_1=cell;
        set[i].slt_1=slot_1;
        set[i].ter=terminal;
        goto step3;
        }
    }

step3:
return delay;

        }
/********************************************************************************/

unsigned long Handoff_2(unsigned long ongoing_call)
 {
 HO_performed_2++;
 unsigned long PLMN_delay=80*ongoing_call;
 unsigned long  delay=0;
 delay=delay+70;                    // HO_Ref_Number (uplink)    //
 delay=delay+70;                    // HO_Detect (downlink)      //
 delay=delay+70;                    // HO_Complete (uplink)      //

 delay=delay+PLMN_delay;              // HO_Complete       //
 delay=delay+PLMN_delay;              // Clear_Command     //
 delay=delay+PLMN_delay;              // Clear_Complete    //

 return delay;
 }

/********************************************************************************/
void call_period_function(int xx, int yy, unsigned long global_timer)
     {
      float period=1000000.0*Exponential_1();
      speech_pkt[xx][yy]=(0.45*period)/pkt_length;
      silence_pkt[xx][yy]=(0.55*period)/pkt_length;
      call_period[xx][yy]=speech_pkt[xx][yy]+silence_pkt[xx][yy];
      next_call[xx][yy]=global_timer+(3220*Poisson_call_50());
     }
/********************************************************************************/
int quality_check_function(int xx, int yy)
    {
     int quality=0;
     long v=speech_pkt[xx][yy]/100;
     if (dropped_pkt[xx][yy]>=v) quality=0;
     if (dropped_pkt[xx][yy]<v)  quality=1;
     if (dropped_pkt[xx][yy]>worst_dropped)  worst_dropped=dropped_pkt[xx][yy];
     return quality;
```

```
    }
/*****************************************************************************/
void call_terminate_function(int xx, int yy)
    {
    burst[xx][yy]='i';
    pkt_ready[xx][yy]=0;
    ack[xx][yy]=0;
    dropped_pkt[xx][yy]=0;
    pkt_exp_timer[xx][yy]=0;
    speech_pkt[xx][yy]=0;
    speech_pkt_count[xx][yy]=0;
    silence_pkt[xx][yy]=0;
    silence_pkt_count[xx][yy]=0;
    call_period[xx][yy]=0;
    next_call[xx][yy]=0;
    next_HO[xx][yy]=0;
    next_stage[xx][yy]=0;
    access_delay[xx][yy]=0;
    max_access_delay[xx][yy]=0;
    HO_delay[xx][yy]=0;
    if (user_slot[xx][yy]!=6000)
        {
        slot[user_slot[xx][yy]]='i';
        user_slot[xx][yy]=6000;
        }
    }
/*****************************************************************************/
char silence_activity_function(unsigned long ww, unsigned long xx, unsigned long yy, unsigned long zz)
   {
   char result;
   int coin_toss=random(20);
   if (coin_toss>0)
   {
          if ((ww-xx)>0)
            {
            result='D';
            goto out_side;
            }
          result='S';
          goto out_side;
   }

   if (coin_toss==0)
   {
          if ((yy-zz)>0)
            {
            result='S';
            goto out_side;
            }
          result='D';
          goto out_side;
   }

out_side:
return result;
}
```

```
/********************************* MAIN **************************************/

void main()
{
    int pkt_buffer=2*slots_per_frame;
    int slot_period=70;
    char result;
    int temp=0;

    unsigned long     ongoing_greatest=0;
    unsigned long     longest=0;
    unsigned long     bad_call=0;
    unsigned long     ongoing_call=0;
    unsigned long     call_finished=0;
    unsigned long     new_call_start=0;
    unsigned long     total_access_delay=0;
    unsigned long     mean_access_delay=0;

    time_t s;
    srand(unsigned (time(&s)));

/******************** FUNCTION TO MAKE IDLE CHANNELS ********************/

unsigned int slot_counter=0;
for (int i=0; i<slots_per_frame; i++)
    {
     for (int j=0; j<total_channels; j++)
         {
         if ((i>1) && (fmod(i+1,3)==0))
             {
             slot[slot_counter]='R';
             goto step_slot;
             }
         slot[slot_counter]='i';
step_slot:
         slot_counter++;
         }
    }


 for(int ii=0; ii<set_limit; ii++)
      {
        set[ii].figure=0;
        set[ii].cell_1=6000;
        set[ii].slt_1=6000;
        set[ii].ter=6000;
      }


for (int r=0; r<7; r++)
    {
     terminal_per_cell[r]=start_up_limit;
     for (int y=0; y<max_terminals; y++)
         {
         burst[r][y]='i';
         pkt_ready[r][y]=0;
         pkt_exp_timer[r][y]=0;
         dropped_pkt[r][y]=0;
         speech_pkt[r][y]=0;
```

```
            speech_pkt_count[r][y]=0;
            silence_pkt[r][y]=0;
            silence_pkt_count[r][y]=0;
            ack[r][y]=0;
            access_delay[r][y]=0;
            max_access_delay[r][y]=0;
            HO_delay[r][y]=0;
            user_slot[r][y]=6000;
            next_stage[r][y]=0;
            next_call[r][y]=0;
            next_HO[r][y]=0;
            call_period[r][y]=0;
            if (y < start_up_limit) call_period_function(r,y,global_timer);
            }
        }

/**************************** SIMULATION START ******************************/

int w=0;                                          /**** w= slot counter ****/
for (long u=0; u<80000; u++)
{
 w=0;                                             /**********************************/
 for (int pp=0; pp<no_of_cells; pp++)             /* Resetting the reservation slots          */
     {                                            /**********************************/
      for (int jj=0; jj<channel_per_cell; jj++)
          {
            reserve[pp][jj]=0;
          }
     }

 for (int x=0; x<slots_per_frame ;x++)
     {
     global_timer=global_timer+slot_period;
     for (int cell=0; cell<no_of_cells; cell++)
         {
          for (int z=0; z<channel_per_cell; z++)
              {

/************************** IF SLOT[W]='i' LOOP  ******************************/

if (slot[w]=='R') goto step_2;

if (slot[w]=='i')
  {
    int highest_prob=0;
    int user_number=0;
    for (int t=0; t<terminal_per_cell[cell]; t++)
     {
      if ((burst[cell][t]=='i') && (global_timer>=next_call[cell][t]) && (next_call[cell][t]>0))
        {
        burst[cell][t]='D';
        pkt_ready[cell][t]=1;
        ack[cell][t]=-1;
        next_call[cell][t]=0;
        new_call_start++;
        if ((cell==0 || cell==3 || cell==4 || cell==5) && (next_HO[cell][t]==0))
            {
             next_HO[cell][t]=global_timer+(1000000*Poisson_15());
            }
```

```
        }

if ((speech_pkt_count[cell][t]+silence_pkt_count[cell][t]>=call_period[cell][t]) &&
(call_period[cell][t]>0))
      {
        if (ack[cell][t]==1) ongoing_call--;
        if (quality_check_function(cell,t)==0) bad_call++;
        total_dropped_pkt=total_dropped_pkt+dropped_pkt[cell][t];
        if (access_delay[cell][t]>max_access_delay[cell][t])
        max_access_delay[cell][t]=access_delay[cell][t];
        if(burst[cell][t]=='H')
         {
           for (int kk=0; kk<set_limit; kk++)
               {
               if ((set[kk].cell_1==cell) && (set[kk].ter==t))
                   {
                   set[kk].figure=0;
                   set[kk].cell_1=6000;
                   set[kk].slt_1=6000;
                   set[kk].ter=6000;
                   goto step_H;
                   }
               }
         }
step_H:

        total_access_delay=total_access_delay+max_access_delay[cell][t];
        call_terminate_function(cell,t);
        call_period_function(cell,t, global_timer);
        call_finished++;
        goto step_1;
      }

    if ((burst[cell][t]=='H') && (ack[cell][t]==1) && (next_stage[cell][t]==1))
    {
    for (int k=0; k<set_limit; k++)
        {
        if ((set[k].cell_1==cell) && (set[k].ter==t))
            {
             if ((global_timer>=set[k].figure) && (set[k].figure>0))
                {
                    if (cell==0)  new_cell=3;
                    if (cell==5)  new_cell=4;
                    if (cell==3)  new_cell=0;
                    if (cell==4)  new_cell=5;
                    int new_terminal=terminal_per_cell[new_cell];
                    burst[new_cell][new_terminal]='H';
                    ck[new_cell][new_terminal]=-1;
                    pkt_ready[new_cell][new_terminal]=1;
                    pkt_exp_timer[new_cell][new_terminal]=0;
                    dropped_pkt[new_cell][new_terminal]=dropped_pkt[cell][t];
                    speech_pkt[new_cell][new_terminal]=speech_pkt[cell][t];
                    speech_pkt_count[new_cell][new_terminal]=speech_pkt_count[cell][t];
                    silence_pkt[new_cell][new_terminal]=silence_pkt[cell][t];
                    silence_pkt_count[new_cell][new_terminal]=silence_pkt_count[cell][t];
                    call_period[new_cell][new_terminal]=call_period[cell][t];
                    access_delay[new_cell][new_terminal]=access_delay[cell][t];
                    max_access_delay[new_cell][new_terminal]=max_access_delay[cell][t];
                    next_stage[new_cell][new_terminal]=2;
```

```
                    call_terminate_function(cell,t);
                    terminal_per_cell[new_cell]++;
                    terminal_per_cell[cell]--;
                    set[k].figure=6000;
                    set[k].cell_1=6000;
                    set[k].slt_1=6000;
                    set[k].ter=6000;
                    goto step_1;

                }
            }
        }
    }



if ((cell==0 || cell==3 || cell==4 || cell==5) && (global_timer>=next_HO[cell][t]) &&
(next_HO[cell][t]>0))
 {
   if ((burst[cell][t]=='D') && (ack[cell][t]==1))
       {
         burst[cell][t]='H';
         HO_delay[cell][t]=HO_delay[cell][t]+Handoff_1(global_timer,w,ongoing_call,cell,t);
         next_HO[cell][t]=0;
         goto step_1;
       }

   if ((burst[cell][t]=='D') && (ack[cell][t]==-1))
       {
         burst[cell][t]='H';
         next_stage[cell][t]=1;
         next_HO[cell][t]=0;
         goto step;
       }

   if ((burst[cell][t]=='S') && (ack[cell][t]==0))
       {
         burst[cell][t]='H';
         ack[cell][t]=-1;
         pkt_ready[cell][t]=1;
         next_stage[cell][t]=1;
         next_HO[cell][t]=0;
         goto step;
       }

 }        /****** if cell=0 & 3 & 4 & 5 Loop ***********/


step:
       if ((pkt_ready[cell][t]==1) && (ack[cell][t]==-1))
           {
             int probability=contention_prob();
             if (probability > highest_prob)
                 {
                 highest_prob=probability;
                 user_number=t;
                 }
           }
```

```
    step_1:
    }                    /******** t  Loop **********/

/******************************** TRANSMISSION LOOP *************************/
if (reserve[cell][z]>=15) goto step_2;

    if (highest_prob>0)
      {
        if (burst[cell][user_number]=='H')
          {
            if (next_stage[cell][user_number]==1)
              {
                ongoing_call++;
                if (ongoing_call>ongoing_greatest) ongoing_greatest=ongoing_call;
    HO_delay[cell][user_number]=HO_delay[cell][user_number]+Handoff_1(global_timer,w,ongoing_call,
cell,user_number);
                ack[cell][user_number]=1;
                pkt_exp_timer[cell][user_number]=0;
                user_slot[cell][user_number]=w;
                slot[w]='H';
                if (access_delay[cell][user_number] > max_access_delay[cell][user_number])
max_access_delay[cell][user_number]=access_delay[cell][user_number];
                access_delay[cell][user_number]=0;
                reserve[cell][z]++;
                goto step_2;
              }

            if (next_stage[cell][user_number]==2)
              {
                HO_delay[cell][user_number]=HO_delay[cell][user_number]+Handoff_2(ongoing_call);
                if (HO_delay[cell][user_number]>longest)  longest=HO_delay[cell][user_number];
                HO_delay[cell][user_number]=0;
              }
          }
        ongoing_call++;
        if (ongoing_call>ongoing_greatest) ongoing_greatest=ongoing_call;
        burst[cell][user_number]='D';
        ack[cell][user_number]=1;
        pkt_exp_timer[cell][user_number]=0;
        user_slot[cell][user_number]=w;
        slot[w]='D';
        if (access_delay[cell][user_number] > max_access_delay[cell][user_number])
max_access_delay[cell][user_number]=access_delay[cell][user_number];
        access_delay[cell][user_number]=0;
        reserve[cell][z]++;
        goto step_2;
      }
    goto step_2;

  }                    /******** if slot[w]=='i'  Loop ********/

/****************************** IF SLOT[W]='D' LOOP  ****************************/
    if (slot[w]=='D')
      {
        for (int t=0; t<terminal_per_cell[cell]; t++)
            {
            if (user_slot[cell][t]==w)
              {
```

```
                    temp=t;
                    goto skip_1;
                    }
                }
        skip_1:

    if ((speech_pkt_count[cell][temp]+silence_pkt_count[cell][temp]>=call_period[cell][temp]) &&
(call_period[cell][temp]>0))
        {
          if (quality_check_function(cell,temp)==0) bad_call++;
          total_dropped_pkt=total_dropped_pkt+dropped_pkt[cell][temp];
          if (access_delay[cell][temp]>max_access_delay[cell][temp])
max_access_delay[cell][temp]=access_delay[cell][temp];
          total_access_delay=total_access_delay+max_access_delay[cell][temp];
          call_terminate_function(cell,temp);
          call_period_function(cell,temp, global_timer);
          ongoing_call--;
          call_finished++;
          goto step_2;
        }
    goto step_2;
    }           /**** if slot[w]=D ****************/

/*************************** IF SLOT[W]='H' LOOP *****************************/

 int set_no;
 int p;
 if (slot[w]=='H')
    {
     for (int k=0; k<set_limit; k++)
        {
         if ((set[k].slt_1==w) && (set[k].cell_1==cell))
            {
             set_no=k;
             p=set[k].ter;
             goto step_M;
            }
        }

  step_M:
    if ((speech_pkt_count[cell][p]+silence_pkt_count[cell][p]>=call_period[cell][p]) &&
(call_period[cell][p]>0))
        {
          if (quality_check_function(cell,p)==0) bad_call++;
          total_dropped_pkt=total_dropped_pkt+dropped_pkt[cell][p];
          if (access_delay[cell][p]>max_access_delay[cell][p])
max_access_delay[cell][p]=access_delay[cell][p];
          set[set_no].figure=0;
          set[set_no].cell_1=6000;
          set[set_no].slt_1=6000;
          set[set_no].ter=6000;
          total_access_delay=total_access_delay+max_access_delay[cell][p];
          call_terminate_function(cell,p);
          call_period_function(cell,p, global_timer);
          ongoing_call--;
          call_finished++;
          goto step_2;
        }
```

```
    if (next_stage[cell][p]==1)
     {
     if (global_timer>=set[set_no].figure)
        {
        if (cell==0)  new_cell=3;
        if (cell==5)  new_cell=4;
        if (cell==3)  new_cell=0;
        if (cell==4)  new_cell=5;
        int new_terminal=terminal_per_cell[new_cell];

        burst[new_cell][new_terminal]='H';
        ack[new_cell][new_terminal]=-1;
        pkt_ready[new_cell][new_terminal]=1;
        pkt_exp_timer[new_cell][new_terminal]=0;
        dropped_pkt[new_cell][new_terminal]=dropped_pkt[cell][p];
        speech_pkt[new_cell][new_terminal]=speech_pkt[cell][p];
        speech_pkt_count[new_cell][new_terminal]=speech_pkt_count[cell][p];
        silence_pkt[new_cell][new_terminal]=silence_pkt[cell][p];
        silence_pkt_count[new_cell][new_terminal]=silence_pkt_count[cell][p];
        call_period[new_cell][new_terminal]=call_period[cell][p];
        access_delay[new_cell][new_terminal]=access_delay[cell][p];
        max_access_delay[new_cell][new_terminal]=max_access_delay[cell][p];
        next_stage[new_cell][new_terminal]=2;
        call_terminate_function(cell,p);
        terminal_per_cell[new_cell]++;
        terminal_per_cell[cell]--;
        set[set_no].figure=0;
        set[set_no].cell_1=6000;
        set[set_no].slt_1=6000;
        set[set_no].ter=6000;
        goto step_2;
        }
     }

  }  /***** if slot[w]='H'    Loop **********/


step_2:
          w=w+1;

 }          /*****  z    channel loop  ******/


/*********************** PACKET_EXPIRY_TIMER UPDATE  ***********************/

for (int k=0; k<terminal_per_cell[cell]; k++)
 {
   char temp_burst, out_come;
   if ((ack[cell][k]==0) && (burst[cell][k]=='i')) goto step_4;
   if ((ack[cell][k]==0) && (burst[cell][k]=='S'))
     {
       silence_pkt_count[cell][k]=silence_pkt_count[cell][k]+1;
     }

   if (ack[cell][k]==-1)
     {
       access_delay[cell][k]=access_delay[cell][k]+slot_period;
       if (access_delay[cell][k] > max_access_delay[cell][k])
       max_access_delay[cell][k]=access_delay[cell][k];
```

```
        if (burst[cell][k]=='H')
          {
            HO_delay[cell][k]=HO_delay[cell][k]+slot_period;
            goto step_4;
          }
        pkt_exp_timer[cell][k]++;
        if (pkt_exp_timer[cell][k]>=pkt_buffer)
          {
            dropped_pkt[cell][k]++;
            pkt_exp_timer[cell][k]=0;
            speech_pkt_count[cell][k]++;
            access_delay[cell][k]=0;
          }
        goto step_4;
      }

    if ((ack[cell][k]==1) && (burst[cell][k]=='D'))   speech_pkt_count[cell][k]++;

    temp_burst=burst[cell][k];
    unsigned long no_1, no_2, no_3, no_4;
    no_1=speech_pkt[cell][k];
    no_2=speech_pkt_count[cell][k];
    no_3=silence_pkt[cell][k];
    no_4=silence_pkt_count[cell][k];
    out_come=silence_activity_function(no_1, no_2, no_3, no_4);

    if ((temp_burst=='D') && (out_come=='D')) goto step_4;
    if ((temp_burst=='S') && (out_come=='S')) goto step_4;
    if ((temp_burst=='D') && (out_come=='S'))
      {
        burst[cell][k]='S';
        pkt_ready[cell][k]=0;
        ack[cell][k]=0;
        slot[user_slot[cell][k]]='i';
        user_slot[cell][k]=6000;
        ongoing_call--;
      }

    if ((temp_burst=='S') && (out_come=='D'))
      {
        burst[cell][k]='D';
        pkt_ready[cell][k]=1;
        ack[cell][k]=-1;
      }

    step_4:
  }           /******* k loop *********/


        }       /**** cell counter loop ****/
    }           /**** x (slot) loop ********/
}               /**** U loop ***************/


if (call_finished>0)
  {
    average_dropped_pkt=total_dropped_pkt/call_finished;
    mean_access_delay=total_access_delay/call_finished;
```

```
}

cout << "  Max ongoing call= " << ongoing_greatest << endl;
cout << "  Max HO_delay= " << longest << endl;
cout << "  HO_performed_1= " << HO_performed_1 << endl;
cout << "  HO_performed_2= " << HO_performed_2 << endl;
cout << "  new_call_start= " << new_call_start << endl;
cout << "  call_finished= " << call_finished << endl;
cout << "  bad_call= " << bad_call << endl;
cout << "  worst_dropped= " << worst_dropped << endl;
cout << "   average_dropped_pkt= " << average_dropped_pkt << endl;
cout << "   mean_access_delay= " <<  mean_access_delay << endl;
cout << "  global_timer= " << global_timer << endl;

}           /*** main function ****/
```

```
/*******************************************************************************/
/*            Advanced   PRMA++                                               */
/*                                                                            */
/*      IMPROVED HANDOVER SIGNALLING USING 3 MESSAGE SEQUENCE ONLY            */
/*                                                                            */
/*   New PRMA operation with Improved Handover function through               */
/*   Air Interface (NO-PRIORITY).                                             */
/*                                                                            */
/*   Standard Specifications:                                                 */
/*   -----------------------                                                  */
/*   PRMA++ Transmission: Packet Size= 125 bits ==> 70 MicroSec               */
/*   Channel Rate= 1.8 Mb/s ==> 0.55 MicroSec/bit                             */
/*  Packet Dropping Threshold= 10 milliSec   (2 frames)                       */
/*   slots per frame= 72                                                      */
/*   Voice encoder= 13 kb/s                                                   */
/*   Acceptable Packet loss Rate= 1 percent or lower                          */
/*                                                                            */
/*   Algorithm Specifications:                                                */
/*   -------------------------                                                */
/*   46 slots per channel (30 Info. & 16 Reservation)                         */
/*                                                                            */
/*    Changable specifiations:                                                */
/*   ----------------------                                                   */
/*     max_terminals= start_up_limit+80                                       */
/*    start_up_limit= 30*2*scaling factor                                     */
/*******************************************************************************/

#include <iostream.h>
#include <fstream.h>
#include <stdlib.h>
#include <iomanip.h>
#include <time.h>
#include <math.h>
#include <alloc.h>
#include <string.h>
#include "my_lib.h"

/************************* SPECIFICATIONS DATA *****************************/
int shift_slots=5;
int channel_per_cell=2;
int slots_per_frame=46;
int no_of_cells=7;
int total_channels=14;
int max_terminals=220;              // max terminals allowed per cell //
int start_up_limit=140;             // start_up no of terminals per cell //
unsigned int ter_per_cell[7];

/*****************************************************************************/
int cell=0;
int hello;
int pkt_length=70;                          /***** packet_length in Microseconds ****/
unsigned long    average_dropped_pkt=0;
unsigned long    HO_performed=0;
unsigned long    worst_dropped=0;
unsigned long     global_timer=0;
unsigned long    total_dropped_pkt=0;
unsigned long    total_dropped_pkt_2=0;
```

```
unsigned int            reserve[7][2];
unsigned  long far      call_period[7][220];
unsigned  long far      dropped_pkt[7][220];
unsigned  long far      access_delay[7][220];
unsigned  long far      HO_delay[7][220];
unsigned  long far      speech_pkt[7][220];
unsigned  long far      silence_pkt[7][220];
unsigned  long far      speech_pkt_count[7][220];
unsigned  long far      silence_pkt_count[7][220];
unsigned  long far      max_access_delay[7][220];

unsigned char far       burst[2][7][220];
unsigned int  far       user_slot[7][220];
unsigned int  far       pkt_ready[7][220];
unsigned int  far       pkt_exp_timer[7][220];
unsigned long far       next_call[7][220];
unsigned long far       next_HO[7][220];
int                     ack[7][220];

int new_cell[4];

unsigned char far slot[644];

int set_limit=600;                  /**** HANDOVER SET_LIMIT ********/

class Pair
    {
     public:
        unsigned long figure;
        unsigned int cell_1;
        unsigned int slot_1;
        unsigned int ter_no_1;
        unsigned int cell_2;
        unsigned int slot_2;
        unsigned int connected;
    };
Pair set[4][600];                   /*********** set[set_limit] ***********/


char file_name_1[]="OUT_1.TXT";
ofstream out_file_1=file_name_1;

char file_name_2[]="OUT_2.TXT";
ofstream out_file_2=file_name_2;

char file_name_3[]="OUT_3.TXT";
ofstream out_file_3=file_name_3;


/****************************************************************************/
int contention_prob()
   {
    int k=random(40);
    return k;
   }
```

```
/*****************************************************************************/
char slot_check(int slot_number_1, int slot_number_2, int shift_slots, int slots_per_frame, int
channel_per_cell, int no_of_cells)
{
    int slot_indicator_1=0;
    int slot_indicator_2=0;
    int adjacent_slot_diff;
    char output;
    int lower;
    int upper;
    adjacent_slot_diff=channel_per_cell*no_of_cells;
    for (int i=0; i<slots_per_frame; i++)
        {
        if ((slot_number_1-(i+1)*adjacent_slot_diff)<0) goto skip_A;
        slot_indicator_1=slot_indicator_1+1;
        }

skip_A:

for (int j=0; j<slots_per_frame; j++)
    {
    if ((slot_number_2-(j+1)*adjacent_slot_diff)<0) goto skip_AA;
    slot_indicator_2=slot_indicator_2+1;
    }

skip_AA:

if ((slot_indicator_1-shift_slots)>=0)
    {
    lower=slot_indicator_1-shift_slots;
    goto skip_B;
    }

if ((slot_indicator_1-shift_slots)<0) lower=slots_per_frame+(slot_indicator_1-shift_slots);

skip_B:

if ((slot_indicator_1+shift_slots)>(slots_per_frame-1))
    {
    upper=(slot_indicator_1+shift_slots)-(slots_per_frame);
    goto skip_C;
    }

if ((slot_indicator_1+shift_slots)<=(slots_per_frame-1)) upper=slot_indicator_1+shift_slots;

skip_C:

if ((slot_indicator_2==slot_indicator_1) || (slot_indicator_2==upper) || (slot_indicator_2==lower))
    {
    output='N';
    goto skip_D;
    }

output='Y';
skip_D:
return output;
}
```

```
/*****************************************************************************/
unsigned int  Handoff_3()
 {
 HO_performed++;
 unsigned  int delay=70*3;              // Measurement Report (ISDN) //
 return delay;
 }
/*****************************************************************************/
void call_period_function(int xx, int yy, unsigned long global_timer)
     {
     float period=1000000.0*Exponential_1();
     speech_pkt[xx][yy]=(0.45*period)/pkt_length;
     silence_pkt[xx][yy]=(0.55*period)/pkt_length;
     call_period[xx][yy]=speech_pkt[xx][yy]+silence_pkt[xx][yy];
     next_call[xx][yy]=global_timer+(3220*Poisson_call_50());
     }

/*****************************************************************************/
int quality_check_function(int xx, int yy)
    {
    int quality=0;
    long v=speech_pkt[xx][yy]/100;
    if (dropped_pkt[xx][yy]>=v) quality=0;
    if (dropped_pkt[xx][yy]<v)  quality=1;
    if (dropped_pkt[xx][yy]>worst_dropped)  worst_dropped=dropped_pkt[xx][yy];
    return quality;
    }
/*****************************************************************************/
void call_terminate_function(int xx, int yy)
     {
     burst[0][xx][yy]='i';
     burst[1][xx][yy]='i';
     pkt_ready[xx][yy]=0;
     ack[xx][yy]=0;
     dropped_pkt[xx][yy]=0;
     pkt_exp_timer[xx][yy]=0;
     speech_pkt[xx][yy]=0;
     speech_pkt_count[xx][yy]=0;
     silence_pkt[xx][yy]=0;
     silence_pkt_count[xx][yy]=0;
     call_period[xx][yy]=0;
     next_call[xx][yy]=0;
     next_HO[xx][yy]=0;
     HO_delay[xx][yy]=0;
     access_delay[xx][yy]=0;
     max_access_delay[xx][yy]=0;
     if (user_slot[xx][yy]!=6000)
         {
         slot[user_slot[xx][yy]]='i';
         user_slot[xx][yy]=6000;
         }
     }

/*****************************************************************************/
char silence_activity_function(unsigned long ww, unsigned long xx, unsigned long yy, unsigned long zz)
     {
     char result;
     int coin_toss=random(25);
     if (coin_toss>0)
```

```
         {
          if ((ww-xx)>0)
              {
               result='D';
               goto out_side;
              }
          result='S';
          goto out_side;
         }

        if (coin_toss==0)
            {
            if ((yy-zz)>0)
                 {
                 result='S';
                 goto out_side;
                 }
            result='D';
            goto out_side;
            }

out_side:
      return result;
     }

/******************************* MAIN ****************************************/
void main()
{

     int pkt_buffer=2*slots_per_frame;
     int slot_period=70;
     int chk_cell;
     int set_cell;
     int set_no;
     int p;
     char result;
     int temp=0;
     int HO_stage=0;
     int highest_prob=0;
     int user_number=0;

     unsigned long     ongoing_greatest=0;
     unsigned long     longest=0;
     unsigned long     bad_call=0;
     unsigned long     bad_call_2=0;
     unsigned long     ongoing_call=0;
     unsigned long     call_finished=0;
     unsigned long     call_finished_2=0;
     unsigned long     new_call_start=0;
     unsigned long     mean_access_delay=0;
     unsigned long     mean_access_delay_2=0;
     unsigned long     total_access_delay=0;
     unsigned long     total_access_delay_2=0;

time_t s;
srand(unsigned (time(&s)));

 new_cell[0]=3;
 new_cell[1]=2;
```

```
 new_cell[2]=1;
 new_cell[3]=0;

/********************** FUNCTION TO MAKE IDLE CHANNELS *******************/

unsigned int slot_counter=0;
for (int i=0; i<slots_per_frame; i++)
    {
     for (int j=0; j<total_channels; j++)
         {
         if ((i>1) && (fmod(i+1,3)==0))
             {
              slot[slot_counter]='R';
              goto step_slot;
             }
         slot[slot_counter]='i';
step_slot:
         slot_counter++;
         }
    }


for(int ii=0; ii<4; ii++)
 {
    for (int jj=0; jj<set_limit; jj++)
        {
        set[ii][jj].figure=0;
        set[ii][jj].cell_1=6000;
        set[ii][jj].slot_1=6000;
        set[ii][jj].ter_no_1=6000;
        set[ii][jj].cell_2=6000;
        set[ii][jj].slot_2=6000;
        set[ii][jj].connected=0;
        }
 }


for (int r=0; r<7; r++)
    {
    ter_per_cell[r]=start_up_limit;
    for (int y=0; y<max_terminals; y++)
         {
         burst[0][r][y]='i';
         burst[1][r][y]='i';
         pkt_ready[r][y]=0;
         pkt_exp_timer[r][y]=0;
         dropped_pkt[r][y]=0;
         speech_pkt[r][y]=0;
         speech_pkt_count[r][y]=0;
         silence_pkt[r][y]=0;
         silence_pkt_count[r][y]=0;
         ack[r][y]=0;
         HO_delay[r][y]=0;
         access_delay[r][y]=0;
         max_access_delay[r][y]=0;
         user_slot[r][y]=6000;
         next_call[r][y]=0;
         next_HO[r][y]=0;
         call_period[r][y]=0;
```

```
        if (y < start_up_limit) call_period_function(r,y,global_timer);
        }
    }

/************************* SIMULATION START **********************************/
unsigned int counter=0;            /**** slot counter ****/
for (long u=0; u<80000; u++)
{
counter=0;
for (int pp=0; pp<no_of_cells; pp++)
    {
     for (int jj=0; jj<channel_per_cell; jj++)
        {
          reserve[pp][jj]=0;
        }
    }

 for (int x=0; x<slots_per_frame ;x++)
    {
    global_timer=global_timer+slot_period;
    for (int cell_counter=0; cell_counter<no_of_cells; cell_counter++)
        {
        cell=cell_counter;
        for (int z=0; z<channel_per_cell; z++)
            {

/*************************  IF SLOT[COUNTER]='i' LOOP  ***********************/

if (slot[counter]=='R') goto step_2;
if (slot[counter]=='i')
 {
    if ((cell==0) || (cell==1) || (cell==2) || (cell==3))
        {
        for (int i=0; i<set_limit; i++)
            {
            if ((set[cell][i].cell_2==cell) && (set[cell][i].connected==0))
                {
                char check=slot_check(set[cell][i].slot_1, counter, shift_slots, slots_per_frame,
channel_per_cell, no_of_cells);
                if (check=='Y')
                    {
                    burst[1][set[cell][i].cell_1][set[cell][i].ter_no_1]='H';
                    set[cell][i].figure=global_timer+Handoff_3();
                    slot[counter]='H';
                    set[cell][i].slot_2=counter;
                    set[cell][i].connected=1;
                    goto step_2;
                    }
                }


    if ((set[cell][i].cell_2==cell) && (set[cell][i].connected==1))
        {
            if ((global_timer>=set[cell][i].figure) && (set[cell][i].figure>0))
                {
                int A_cell=set[cell][i].cell_1;
                int A_ter_no=set[cell][i].ter_no_1;
                HO_delay[A_cell][A_ter_no]=HO_delay[A_cell][A_ter_no]+Handoff_3();
                unsigned long temp_delay=HO_delay[A_cell][A_ter_no];
```

```
                    if (temp_delay>longest) longest=temp_delay;
                    burst[0][A_cell][A_ter_no]='i';
                    burst[1][A_cell][A_ter_no]='i';
                    burst[0][cell][ter_per_cell[cell]]='D';
                    ack[cell][ter_per_cell[cell]]=1;
                    pkt_ready[cell][ter_per_cell[cell]]=1;
                    user_slot[cell][ter_per_cell[cell]]=set[cell][i].slot_2;
                    slot[set[cell][i].slot_2]='D';
                    speech_pkt[cell][ter_per_cell[cell]]=speech_pkt[A_cell][A_ter_no];
                    speech_pkt_count[cell][ter_per_cell[cell]]=speech_pkt_count[A_cell][A_ter_no];
                    silence_pkt[cell][ter_per_cell[cell]]=silence_pkt[A_cell][A_ter_no];
                    silence_pkt_count[cell][ter_per_cell[cell]]=speech_pkt_count[A_cell][A_ter_no];
                    call_period[cell][ter_per_cell[cell]]=call_period[A_cell][A_ter_no];
                    dropped_pkt[cell][ter_per_cell[cell]]=dropped_pkt[A_cell][A_ter_no];
                    access_delay[cell][ter_per_cell[cell]]=access_delay[A_cell][A_ter_no];
                    max_access_delay[cell][ter_per_cell[cell]]=max_access_delay[A_cell][A_ter_no];
                    call_terminate_function(A_cell,A_ter_no);
                    ter_per_cell[cell]++;
                    ter_per_cell[A_cell]--;
                    set[cell][i].figure=0;
                    set[cell][i].cell_1=6000;
                    set[cell][i].slot_1=6000;
                    set[cell][i].ter_no_1=6000;
                    set[cell][i].cell_2=6000;
                    set[cell][i].slot_2=6000;
                    set[cell][i].connected=0;
                    }
                }
            }           /******** for i<set_limit loop *********/
        }               /******** if cell=0 || 1 || 2 || 3 **********/

/*********************** CALLS GENERATION FUNCTION ***********************/
 highest_prob=0;
 user_number=0;

for (int t=0; t<ter_per_cell[cell]; t++)
    {
    if ((burst[0][cell][t]=='i') && (global_timer>=next_call[cell][t]) && (next_call[cell][t]>0))
        {
        burst[0][cell][t]='D';
        pkt_ready[cell][t]=1;
        ack[cell][t]=-1;
        next_call[cell][t]=0;
        new_call_start++;
        if ((cell==0 || cell==1 || cell==2 || cell==3) && (next_HO[cell][t]==0))
            {
            next_HO[cell][t]=global_timer+(1000000*Poisson_15());
            }
        goto step;
        }

    if ((speech_pkt_count[cell][t]+silence_pkt_count[cell][t]>=call_period[cell][t]) &&
(call_period[cell][t]>0))
        {
            if (ack[cell][t]==1) ongoing_call--;
            if (quality_check_function(cell,t)==0) bad_call++;
            if (cell==0 || cell==1 || cell==2 || cell==3)  bad_call_2++;
            total_dropped_pkt=total_dropped_pkt+dropped_pkt[cell][t];
            if (access_delay[cell][t] > max_access_delay[cell][t])
```

```
            max_access_delay[cell][t]=access_delay[cell][t];
            if(burst[0][cell][t]=='T')
                {
                for (int kk=0; kk<set_limit; kk++)
                    {
                    if ((set[new_cell[cell]][kk].cell_1==cell) && (set[new_cell[cell]][kk].ter_no_1==t))
                        {
                        set[cell][i].figure=0;
                        set[cell][i].cell_1=6000;
                        set[cell][i].slot_1=6000;
                        set[cell][i].ter_no_1=6000;
                        set[cell][i].cell_2=6000;
                        set[cell][i].slot_2=6000;
                        set[cell][i].connected=0;
                        goto step_H;
                        }
                    }
                }
step_H:

        total_access_delay=total_access_delay+max_access_delay[cell][t];
        if (cell==0 || cell==1 || cell==2 || cell==3)
            {
            total_access_delay_2=total_access_delay_2+max_access_delay[cell][t];
            call_finished_2++;
            }
        call_terminate_function(cell,t);
        call_period_function(cell,t, global_timer);
        call_finished++;
        goto step_1;
    }

/******************** HANDOVER GENERATION FUNCTION **********************/

if (((cell==0) || (cell==1) || (cell==2) || (cell==3)) && (global_timer>=next_HO[cell][t]) &&
(next_HO[cell][t]>0))
    {
        if ((burst[0][cell][t]=='D') && (ack[cell][t]==1))
        {
            burst[0][cell][t]='T';
            next_HO[cell][t]=0;
            slot[user_slot[cell][t]]='T';

            /***** find an empty set in the next_cell ******/

            for (int j=0; j<set_limit; j++)
                {
                if (set[new_cell[cell]][j].cell_1==6000)
                    {
                    set[new_cell[cell]][j].figure=0;
                    set[new_cell[cell]][j].cell_1=cell;
                    set[new_cell[cell]][j].slot_1=user_slot[cell][t];
                    set[new_cell[cell]][j].ter_no_1=t;
                    set[new_cell[cell]][j].cell_2=new_cell[cell];
                    set[new_cell[cell]][j].slot_2=6000;
                    set[new_cell[cell]][j].connected=0;
                    goto step_1;
                    }
                }
```

```
            }

        if ((burst[0][cell][t]=='S') && (ack[cell][t]==0))
            {
            burst[0][cell][t]='T';
            ack[cell][t]=-1;
            pkt_ready[cell][t]=1;
            next_HO[cell][t]=0;
            goto step;
            }

        if ((burst[0][cell][t]=='D') && (ack[cell][t]==-1))
            {
            burst[0][cell][t]='T';
            next_HO[cell][t]=0;
            goto step;
            }

    }           /******* if cell=0 || 1 || 2 || 3  ************/
step:

        if ((pkt_ready[cell][t]==1) && (ack[cell][t]==-1))
            {
            int probability=contention_prob();
            if (probability > highest_prob)
                {
                highest_prob=probability;
                user_number=t;
                }
                goto step_1;
            }

    step_1:
 }              /********** t  Loop ******************/

/*************************** TRANSMISSION LOOP  ***************************/

if (reserve[cell][z]>=15) goto step_2;

    if (highest_prob>0)
    {
    if (burst[0][cell][user_number]=='T')
        {
        for (int j=0; j<set_limit; j++)
            {
            if (set[new_cell[cell]][j].cell_1==6000)
                {
                set[new_cell[cell]][j].figure=0;
                set[new_cell[cell]][j].cell_1=cell;
                set[new_cell[cell]][j].slot_1=counter;
                set[new_cell[cell]][j].ter_no_1=user_number;
                set[new_cell[cell]][j].cell_2=new_cell[cell];
                set[new_cell[cell]][j].slot_2=6000;
                set[new_cell[cell]][j].connected=0;
                goto step_T;
                }
            }
step_T:
        ack[cell][user_number]=1;
```

```
            pkt_ready[cell][user_number]=1;
            pkt_exp_timer[cell][user_number]=0;
            user_slot[cell][user_number]=counter;
            slot[counter]='T';
            if (access_delay[cell][user_number] > max_access_delay[cell][user_number])
            max_access_delay[cell][user_number]=access_delay[cell][user_number];
            access_delay[cell][user_number]=0;
            ongoing_call++;
            if (ongoing_call>ongoing_greatest) ongoing_greatest=ongoing_call;
            reserve[cell][z]++;
            goto step_2;
        }

            burst[0][cell][user_number]='D';
            ack[cell][user_number]=1;
            pkt_ready[cell][user_number]=1;
            pkt_exp_timer[cell][user_number]=0;
            user_slot[cell][user_number]=counter;
            slot[counter]='D';
            if (access_delay[cell][user_number] > max_access_delay[cell][user_number])
            max_access_delay[cell][user_number]=access_delay[cell][user_number];
            access_delay[cell][user_number]=0;
            ongoing_call++;
            if (ongoing_call>ongoing_greatest) ongoing_greatest=ongoing_call;

            reserve[cell][z]++;
            goto step_2;
    }

}               /********** if slot[counter]=='i'  Loop **********/

/************************* IF SLOT[counter]='D' LOOP ***************************/

if (slot[counter]=='D')
    {
    for (int t=0; t<ter_per_cell[cell]; t++)
        {
        if (user_slot[cell][t]==counter)
            {
            temp=t;
            goto skip_1;
            }
        }
    skip_1:

if ((speech_pkt_count[cell][temp]+silence_pkt_count[cell][temp]>=call_period[cell][temp]) &&
(call_period[cell][temp]>0))
    {
    if (quality_check_function(cell,temp)==0) bad_call++;
    if (cell==0 || cell==1 || cell==2 || cell==3)  bad_call_2++;
    total_dropped_pkt=total_dropped_pkt+dropped_pkt[cell][temp];
    if (access_delay[cell][temp] > max_access_delay[cell][temp])
    max_access_delay[cell][temp]=access_delay[cell][temp];
    total_access_delay=total_access_delay+max_access_delay[cell][temp];
    if (cell==0 || cell==1 || cell==2 || cell==3)
        {
        total_access_delay_2=total_access_delay_2+max_access_delay[cell][temp];
        call_finished_2++;
        }
```

```
    call_terminate_function(cell,temp);
    call_period_function(cell,temp, global_timer);
    ongoing_call--;
    call_finished++;
    goto step_2;
    }
goto step_2;
}                       /******* end of slot[counter]=D loop ***********/


/********************** IF SLOT[counter]='H' LOOP ****************************/

if ((slot[counter]=='H') || (slot[counter]=='T'))
    {
    for (int h=0; h<set_limit; h++)
        {
        if ((slot[counter]=='T') && (set[new_cell[cell]][h].slot_1==counter) &&
(set[new_cell[cell]][h].cell_1==cell))
        {
        set_no=h;
        p=set[new_cell[cell]][h].ter_no_1;
        set_cell=new_cell[cell];
        chk_cell=cell;
        goto step_CHK;
        }

    if ((slot[counter]=='H') && (set[cell][h].slot_2==counter) && (set[cell][h].cell_2==cell))
        {
        set_no=h;
        p=set[cell][h].ter_no_1;
        set_cell=cell;
        chk_cell=set[cell][h].cell_1;
        goto step_CHK;
        }
    }       /******* end of  h < set_limit loop ******/


step_CHK:

        if ((speech_pkt_count[chk_cell][p]+silence_pkt_count[chk_cell][p]>=call_period[chk_cell][p]) &&
(call_period[chk_cell][p]>0))
            {
            if (quality_check_function(chk_cell,p)==0) bad_call++;
            if (chk_cell==0 || chk_cell==1 || chk_cell==2 || chk_cell==3)  bad_call_2++;
            total_dropped_pkt=total_dropped_pkt+dropped_pkt[chk_cell][p];
            if (access_delay[chk_cell][p] > max_access_delay[chk_cell][p])
            max_access_delay[chk_cell][p]=access_delay[chk_cell][p];
            set[set_cell][set_no].figure=0;
            set[set_cell][set_no].cell_1=6000;
            set[set_cell][set_no].slot_1=6000;
            set[set_cell][set_no].ter_no_1=6000;
            set[set_cell][set_no].cell_2=6000;
            set[set_cell][set_no].slot_2=6000;
            set[set_cell][set_no].connected=0;
            total_access_delay=total_access_delay+max_access_delay[chk_cell][p];
            if (cell==0 || cell==1 || cell==2 || cell==3)
                {
                total_access_delay_2=total_access_delay_2+max_access_delay[chk_cell][p];
```

```
                    call_finished_2++;
                    }

                call_terminate_function(chk_cell,p);
                call_period_function(chk_cell,p, global_timer);
                ongoing_call--;
                call_finished++;
                goto step_2;
                }

    step_M:

      if (set[set_cell][set_no].connected==1)
           {
           if ((global_timer>=set[set_cell][set_no].figure) && (set[set_cell][set_no].figure>0))
                {
                int xxx_1=set[set_cell][set_no].cell_1;
                int yyy_1=set[set_cell][set_no].slot_1;
                int zzz_1=set[set_cell][set_no].ter_no_1;
                int xxx_2=set[set_cell][set_no].cell_2;
                int yyy_2=set[set_cell][set_no].slot_2;
                HO_delay[xxx_1][zzz_1]=HO_delay[xxx_1][zzz_1]+Handoff_3();
                unsigned long temp_delay=HO_delay[xxx_1][zzz_1];
                if (temp_delay>longest) longest=temp_delay;
                burst[0][xxx_1][zzz_1]='i';
                burst[1][xxx_1][zzz_1]='i';
                burst[0][set_cell][ter_per_cell[xxx_2]]='D';
                ack[set_cell][ter_per_cell[xxx_2]]=1;
                pkt_ready[set_cell][ter_per_cell[xxx_2]]=1;
                user_slot[set_cell][ter_per_cell[xxx_2]]=set[set_cell][set_no].slot_2;
                slot[set[set_cell][set_no].slot_2]='D';
                speech_pkt[set_cell][ter_per_cell[xxx_2]]=speech_pkt[xxx_1][zzz_1];
                speech_pkt_count[set_cell][ter_per_cell[xxx_2]]=speech_pkt_count[xxx_1][zzz_1];
                silence_pkt[set_cell][ter_per_cell[xxx_2]]=silence_pkt[xxx_1][zzz_1];
                silence_pkt_count[set_cell][ter_per_cell[xxx_2]]=speech_pkt_count[xxx_1][zzz_1];
                call_period[set_cell][ter_per_cell[xxx_2]]=call_period[xxx_1][zzz_1];
                dropped_pkt[set_cell][ter_per_cell[xxx_2]]=dropped_pkt[xxx_1][zzz_1];
                access_delay[set_cell][ter_per_cell[xxx_2]]=access_delay[xxx_1][zzz_1];
                max_access_delay[set_cell][ter_per_cell[xxx_2]]=max_access_delay[xxx_1][zzz_1];
                call_terminate_function(xxx_1,zzz_1);
                ter_per_cell[xxx_2]++;
                ter_per_cell[xxx_1]--;
                set[set_cell][set_no].figure=0;
                set[set_cell][set_no].cell_1=6000;
                set[set_cell][set_no].slot_1=6000;
                set[set_cell][set_no].ter_no_1=6000;
                set[set_cell][set_no].cell_2=6000;
                set[set_cell][set_no].slot_2=6000;
                set[set_cell][set_no].connected=0;
                goto step_2;
                }
           }

    }                    /***** if slot[counter]='H'   Loop **********/


    step_2:
                counter=counter+1;
```

```
}           /***** z   channel loop ******/

/******************** PACKET_EXPIRY_TIMER UPDATE ************************/

for (int k=0; k<ter_per_cell[cell]; k++)
{
    char temp_burst, out_come;
    if ((burst[0][cell][k]=='i') && (ack[cell][k]==0)) goto step_4;
    if ((burst[0][cell][k]=='S') && (ack[cell][k]==0))
        {
        silence_pkt_count[cell][k]=silence_pkt_count[cell][k]+1;
        }

    if (burst[0][cell][k]=='T')
        {
        if (ack[cell][k]==-1)
            {
            HO_delay[cell][k]=HO_delay[cell][k]+slot_period;
            goto step_4;
            }
        for (int jj=0; jj<set_limit; jj++)
            {
            if ((set[new_cell[cell]][jj].cell_1==cell) && (set[new_cell[cell]][jj].ter_no_1==k))
                {
                if  (set[new_cell[cell]][jj].connected==0)
                    {
                    HO_delay[cell][k]=HO_delay[cell][k]+slot_period;
                    goto step_4;
                    }
                goto step_4;
                }
            }
    }

    if ((burst[0][cell][k]=='D') && (ack[cell][k]==-1))
    {
        pkt_exp_timer[cell][k]++;
        access_delay[cell][k]=access_delay[cell][k]+slot_period;
        if (access_delay[cell][k] > max_access_delay[cell][k])
        max_access_delay[cell][k]=access_delay[cell][k];
        if (pkt_exp_timer[cell][k]>=pkt_buffer)
            {
            dropped_pkt[cell][k]++;
            pkt_exp_timer[cell][k]=0;
            speech_pkt_count[cell][k]++;
            access_delay[cell][k]=0;
            }
        goto step_4;
    }

    if ((burst[0][cell][k]=='D') && (ack[cell][k]==1))   speech_pkt_count[cell][k]++;

    temp_burst=burst[0][cell][k];
    unsigned long no_1, no_2, no_3, no_4;
    no_1=speech_pkt[cell][k];
    no_2=speech_pkt_count[cell][k];
    no_3=silence_pkt[cell][k];
    no_4=silence_pkt_count[cell][k];
```

```
        out_come=silence_activity_function(no_1, no_2, no_3, no_4);

        if ((temp_burst=='D') && (out_come=='D')) goto step_4;
        if ((temp_burst=='S') && (out_come=='S')) goto step_4;

        if ((temp_burst=='D') && (out_come=='S'))
            {
            burst[0][cell][k]='S';
            pkt_ready[cell][k]=0;
            ack[cell][k]=0;
            slot[user_slot[cell][k]]='i';
            user_slot[cell][k]=6000;
            ongoing_call--;
        }

        if ((temp_burst=='S') && (out_come=='D'))
            {
            burst[0][cell][k]='D';
            pkt_ready[cell][k]=1;
            ack[cell][k]=-1;
            }

        step_4:
            }           /******* k loop *********/


            }       /**** cell counter loop ****/
        }       /**** x (slot) loop ********/
    }       /**** U loop ***************/

    if (call_finished>0)
        {
        average_dropped_pkt=total_dropped_pkt/call_finished;
        mean_access_delay=total_access_delay/call_finished;
        }

    if (call_finished_2>0)
        {
        mean_access_delay_2=total_access_delay_2/call_finished_2;
        }

    cout << "      Max ongoing call= " << ongoing_greatest << endl;
    cout << "      Max HO_delay= " << longest << endl;
    cout << "      HO_performed= " << HO_performed << endl;
    cout << "      new_call_start= " << new_call_start << endl;
    cout << "      call_finished= " << call_finished << endl;
    cout << "      call_finished_HO= " << call_finished_2 << endl;
    cout << "      bad_call_total= " << bad_call << endl;
    cout << "      bad_call_HO= " << bad_call_2 << endl;
    cout << "      worst_dropped= " << worst_dropped << endl;
    cout << "      average_dropped_pkt= " << average_dropped_pkt << endl;
    cout << "      mean_access_delay= " << mean_access_delay << endl;
    cout << "      mean_access_delay_HO= " << mean_access_delay_2 << endl;
    cout << "      global_timer= " << global_timer << endl;


    }    /***************** main function ********************/
```

Published and Presented Papers

- Ali; A. H. & Brewster; R. L. "*Handover Signalling for microcellular environments.*", IEE Electronics Letters, pp. 1754-1755, Sep.12th, Vol.32, No.19, 1996.

- Ali; A. H. " *Multi-layered and Integrated Cellular Communication Systems.*" Presented.at the IEE Young Members Short Papers Evening. Feb. 12th, 1996, England.

- Ali; A. H. " *Efficient Handover Signalling for Microcellular Environments using Packet Access Mechanism.*" Presented at the *IEE Propagation & Antennas* Colloquium, London, Feb. 23rd, 1998.