

**Quality of Service Provisioning  
in Cellular Mobile Networks**

by

**FEI YU**

**M.A.Sc., Beijing University of Posts and Telecommunications, 1998**

**B.A.Sc., Dalian University of Technologies, 1995**

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENTS FOR THE DEGREE OF  
DOCTOR OF PHILOSOPHY**

in

**THE FACULTY OF GRADUATE STUDIES  
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING**

**We accept this thesis as conforming  
to the required standard**

**THE UNIVERSITY OF BRITISH COLUMBIA**

**Oct. 2003**

**© Fei Yu, 2003**

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Electrical and Computer Engineering

The University of British Columbia  
Vancouver, Canada

Date Oct. 7, 2003

## Abstract

In recent years, tremendous growth has occurred in the use of cellular mobile communications around the world. This thesis studies Quality of Service (QoS) provisioning in cellular mobile networks. In order to guarantee the handoff dropping probability of mobile users in cellular networks, call admission control and bandwidth reservation schemes are proposed based on more realistic assumptions than those made in existing proposals. A mobility prediction scheme is derived from data compression techniques that are both theoretically optimal and good in practice. In order to utilize resources more efficiently, the proposed scheme predicts not only *which* cell the mobile will handoff to but also *when* the handoff will occur. Based on the mobility prediction, bandwidth is reserved to guarantee a given target handoff dropping probability. Simulation results show that the proposed schemes meet our design goals and outperform the static reservation scheme.

We also develop a framework of combining QoS provisioning and mobility management in cellular mobile networks. The key component of this framework is a common mobility prediction scheme, which can be used in both locating/paging mobiles and in making admission decisions. Novel QoS provisioning and mobility management schemes are proposed in this framework. The performance of the proposed schemes is evaluated using simulations.

Adaptive multimedia applications that can operate over a wide range of available bandwidths are expected to be used in future cellular mobile networks. We present a QoS provisioning scheme in adaptive multimedia cellular networks via reinforcement learning. The proposed scheme does not require explicit state transition probabilities, and therefore, the underlying assumptions of this scheme are more realistic than those in previous schemes. Simulation results

demonstrate the superior performance of the proposed scheme over some of the existing methods.

Finally, call admission control for cellular-to-Internet protocol (IP) internetworking is studied. In order to provide QoS to the Internet and avoid scalability problems, several recent papers propose endpoint measurement-based admission control (EMAC) scheme. Although EMAC has many desirable features in the wireline networks as shown in previous work, in this thesis, we show that several distinct characteristics in cellular mobile networks make EMAC difficult to implement. A novel mobile-EMAC (M-EMAC) scheme for cellular-to-IP internetworking is proposed. Simulation results show that M-EMAC outperforms EMAC in cellular mobile networks.

# Table of Contents

<b>Abstract</b>	<b>ii</b>
<b>List of Figures</b>	<b>viii</b>
<b>List of Tables</b>	<b>xi</b>
<b>Glossary</b>	<b>xii</b>
<b>Acknowledgements</b>	<b>xvii</b>
<b>Chapter 1 Introduction</b>	<b>1</b>
1.1 QoS Provisioning in Cellular Mobile Networks.....	1
1.2 Overview of Related Work.....	4
1.2.1 Call Admission Control for Non-Adaptive Traffic.....	4
1.2.2 Mobility Management.....	5
1.2.3 Call Admission Control and Bandwidth Adaptation for Adaptive Traffic.....	6
1.2.4 QoS Provisioning in the Internet.....	7
1.3 Motivations .....	8
1.4 Main Contributions .....	11
1.5 Organization of the Thesis .....	13
<b>Chapter 2 Mobility-Based Call Admission Control and Bandwidth Reservation</b>	<b>14</b>
2.1 Introduction.....	14
2.2 Model Description .....	14
2.2.1 Network Topology.....	15
2.2.2 Channel Holding Time.....	16
2.2.3 User Mobility Pattern.....	17
2.3 Mobility Prediction.....	18

2.3.1	Optimal Data Compression.....	19
2.3.2	Mobility Prediction .....	21
2.3.3	Analysis of the Mobility Prediction Scheme .....	22
2.3.4	Implementation Considerations .....	25
2.4	Call Admission Control and Bandwidth Reservation.....	27
2.4.1	Calculation of $P(i, j, T_k)$ .....	27
2.4.2	The Most Likely Cell-Time .....	28
2.4.3	Bandwidth Reservation.....	29
2.4.4	Call Admission Control and Bandwidth Reservation for New Calls .....	30
2.4.5	Adaptive Control of Admission Threshold.....	30
2.4.6	Call Admission Control and Bandwidth Reservation for Handoff Calls.....	31
2.5	Simulation Results and Discussions .....	31
2.6	Summary.....	35
<b>Chapter 3</b>	<b>A Framework Combining QoS Provisioning and Mobility Management</b>	<b>39</b>
3.1	Introduction.....	39
3.2	Common Mobility Prediction Scheme .....	39
3.3	Mobility Management in the Combined Framework.....	42
3.3.1	The Original Path-Based Scheme .....	42
3.3.2	The New Path-Based Scheme.....	43
3.3.3	Simulation Results and Discussions .....	45
3.4	Call Admission Control in the Combined Framework .....	46
3.4.1	Time Interval Prediction .....	47
3.4.2	Call Admission Control Scheme.....	50

3.4.3	Simulation Results and Discussions .....	52
3.5	Summary .....	54
<b>Chapter 4</b>	<b>QoS Provisioning for Adaptive Multimedia</b>	<b>60</b>
4.1	Introduction .....	60
4.2	QoS Provisioning in Adaptive Framework .....	62
4.2.1	Adaptive Multimedia Applications .....	62
4.2.2	Adaptive Cellular Mobile Networks .....	63
4.2.3	QoS Provisioning in Adaptive Framework .....	64
4.2.4	Average Reward RL .....	66
4.3	Average Reward RL for Solving SMDP .....	67
4.3.1	Average Reward SMDP .....	67
4.3.2	Solving the Average Reward SMDP Using RL .....	69
4.4	Formulation of QoS Provisioning in Adaptive Framework .....	72
4.4.1	State, Actions and Rewards .....	72
4.4.2	Constraints .....	75
4.4.3	Exploration .....	78
4.4.4	Trading off Action Space Complexity with State Space Complexity .....	79
4.5	Implementation Considerations .....	81
4.5.1	Approximate Representation of Action Values .....	81
4.5.2	Structure and Pseudo-code .....	82
4.6	Simulation Results and Discussions .....	83
4.7	Summary .....	90
<b>Chapter 5</b>	<b>M-EMAC for Cellular-to-IP Internetworking</b>	<b>99</b>

5.1	Introduction.....	99
5.2	EMAC Scheme for the Internet .....	100
5.3	EMAC in Cellular Mobile Networks .....	101
5.3.1	Low Bandwidth.....	102
5.3.2	User Mobility .....	107
5.3.3	High Bit Error Rate.....	109
5.4	Mobile-EMAC for Cellular-to-IP Internetworking.....	110
5.4.1	UMTS QoS Management Functions.....	111
5.4.2	Mobile-EMAC Scheme .....	112
5.4.3	Performance Evaluation.....	114
5.5	Summary .....	116
<b>Chapter 6</b>	<b>Conclusions</b>	<b>119</b>
6.1	Summary.....	119
6.2	Further Work.....	121
	<b>Bibliography</b>	<b>123</b>

## List of Figures

Figure 2.1	Modeling an actual cellular network.....	16
Figure 2.2	The trie constructed in Example 2.1 .....	20
Figure 2.3	A pseudo-code of the mobility prediction scheme .....	22
Figure 2.4	A mobility trie used for mobility prediction .....	23
Figure 2.5	Implementation of trie nodes in Figure 2.2: (a) array and (b) linked list...26	
Figure 2.6	$P_{nb}$ and $P_{hd}$ vs. offered load.....	36
Figure 2.7	Utilization vs. offered load .....	36
Figure 2.8	$P_{hd}$ vs. simulation time with different values of time slot duration.....	37
Figure 2.9	Utilization vs. offered load with different values of time slot duration .....	37
Figure 2.10	Comparison with static reservation (4 BUs).....	38
Figure 2.11	Comparison with static reservation (5 BUs).....	38
Figure 3.1	The trie constructed in Example 3.1 .....	41
Figure 3.2	The original path-based scheme.....	43
Figure 3.3	The proposed path-based scheme .....	44
Figure 3.4	Location update performance gain vs. call-to-mobility ratio.....	56
Figure 3.5	Paging performance gain vs. call-to-mobility ratio .....	56
Figure 3.6	Hyperbolic positioning in E-OTD, $N = 3$ .....	57
Figure 3.7	$P_{nb}$ and $P_{hd}$ vs. offered load.....	57
Figure 3.8	$P_{hd}$ vs. simulation time with different values of adaptive factor .....	58
Figure 3.9	Comparison with YL01 and one of OKS98 schemes .....	58
Figure 3.10	Comparison with YL01 and one of OKS98 schemes .....	59
Figure 3.11	$P_{hd}$ vs. simulation time in YL01 and proposed scheme .....	59

Figure 4.1	Radio bearer reconfiguration in UMTS .....	63
Figure 4.2	A reinforcement learning model .....	70
Figure 4.3	The neural network used in approximation .....	81
Figure 4.4	The structure of the QoS provisioning scheme.....	82
Figure 4.5	A pseudo-code of the QoS provisioning scheme.....	83
Figure 4.6	A cellular network used in simulations.....	84
Figure 4.7	Normalized average rewards.....	91
Figure 4.8	Normalized average rewards vs. adaptation operation cost.....	91
Figure 4.9	Handoff dropping probabilities.....	92
Figure 4.10	New call blocking probabilities of class 1 calls.....	92
Figure 4.11	New call blocking probabilities of class 2 calls.....	93
Figure 4.12	Normalized average bandwidths of class 1 calls .....	93
Figure 4.13	Normalized average bandwidths of class 2 calls .....	94
Figure 4.14	Normalized bandwidths variance of class 1 calls .....	94
Figure 4.15	Normalized bandwidths variance of class 2 calls .....	95
Figure 4.16	Normalized average rewards for different average bandwidth requirements.....	95
Figure 4.17	Normalized average rewards for different bandwidth variance requirements.....	96
Figure 4.18	Normalized average rewards with non-uniform traffic .....	96
Figure 4.19	A traffic pattern of a typical business day .....	97
Figure 4.20	Normalized average rewards with time varying traffic .....	97
Figure 4.21	Normalized average rewards with convex reward function and non-uniform traffic.....	98
Figure 5.1	EMAC scheme for the Internet .....	100

Figure 5.2	Out-of-band probing in the wireless model .....	105
Figure 5.3	Out-of-band probing in the wireline model .....	106
Figure 5.4	In-band probing in the wireless model.....	107
Figure 5.5	In-band probing in the wireline model.....	108
Figure 5.6	Retransmissions of RLC blocks.....	110
Figure 5.7	QoS management functions for the UMTS bearer service in control plane.....	112
Figure 5.8	Mobile-EMAC .....	113
Figure 5.9	Mobile-EMAC procedure .....	114
Figure 5.11	Loss-load curves of M-EMAC and EMAC (probing 20 s).....	118
Figure 5.10	Loss-load curves of M-EMAC and EMAC (probing 5 s).....	118

## List of Tables

Table 4.1	Experimental parameters .....	84
Table 4.2	Number of numerical operations .....	90
Table 5.1	Simulation parameters of traffic source 1 .....	103
Table 5.2	Simulation parameters of traffic source 2 .....	103

# Glossary

## Acronyms

3G - 3rd Generation

3GPP - 3rd Generation Partnership Project

BA - Bandwidth Adaptation

BER - Bit Error Rate

BS - Bearer Service

BTS - Base Transceiver Station

BU - Bandwidth Unit

CAC - Call Admission Control

EDGE - Enhanced Data Rate for GSM Evolution

EMAC - Endpoint Measurement Based Admission Control

E-OTD - Enhanced Observed Time Difference

GPRS - General Packet Radio Service

GSM - Global System for Mobile communications

GTD - Geometric Time Difference

ID - Identifier

IP - Internet Protocol

LA - Location Area

M-EMAC - Mobile Endpoint Measurement Based Admission Control

MLCT - Most Likely Cell-Time

MM - Mobility Management

OTD - Observed Time Difference

PPM - prediction by partial match

QoS - Quality of Service

RL - Reinforcement Learning

RTD - Real Time Difference

SMART - Semi-Markov Average Reward Technique

SMDP - Semi-Markov Decision Process

TCP - Transport Control Protocol

TDOA - Time Difference of Arrival

UMTS - Universal Mobile Telecommunications System

## Symbols

$\alpha$  - The admission threshold

$\varepsilon$  - A design parameter to control  $\alpha$  adaptively

$\lambda$  - New call arrival rate

$\sigma_n$  - The time of the  $(n+1)$ th decision epoch

$\tau_n$  - The time between the  $n$ th and the  $(n+1)$ th decision epochs

$\omega$  - Lagrange multiplier

$\rho_C(x^n)$  - The compression ratio attained by  $C$  for  $x^n$

$AB^i$  - The normalized average allocated bandwidth of class  $i$  calls

$B_r(i, j, m, T_k)$  - The reserved bandwidth in cell  $j$  during  $T_k$  for  $m$  from cell  $i$

$B_r(i, j, m, t_a, t_d)$  - The reserved bandwidth in cell  $j$  during time interval  $t_a$  and  $t_d$

$B_r(j, T_k)$  - The aggregate reserved bandwidth in cell  $j$  during  $T_k$

$B_r(j, t_a, t_d)$  - The aggregate reserved bandwidth in cell  $j$  during time interval  $t_a$  and  $t_d$

$B_f(j, T_k)$  - The free bandwidth in cell  $j$  during  $T_k$

$B_f(j, t_a, t_d)$  - The free bandwidth in cell  $j$  during time interval  $t_a$  and  $t_d$

$C(x^n)$  - The length, in bits, of the output sequence of  $x^n$  after compression

$P_{hd}$  - Handoff dropping probability

$P_{i,j}(T_k)$  - The probability that a mobile in cell  $i$  will visit cell  $j$  during time slot  $T_k$

$P(i, j, t_a, t_d)$  - The probability that a mobile in cell  $i$  will visit cell  $j$  during time interval  $t_a$  and  $t_d$

$P_{nb}$  - New call blocking probability

$P_{ss'}(a)$  - The probability of transition from state  $s$  to state  $s'$  under action  $a$

$P_v$  - The percentage of voice in the offered traffic

$q(x^n)$  - A lower bound on the compression ratio attainable for  $x^n$  by any codebook

$r_{act}(s', s, a)$  - The actual reward between decision epochs

$R(s, a)$  - The action value of doing an action  $a$  in state  $s$

$t_a(i, j)$  - The time when a mobile in cell  $i$  will arrive at cell  $j$

$t_d(i, j)$  - The time when a mobile in cell  $i$  will depart from cell  $j$

$TAB^i$  - The target normalized average allocated bandwidth of class  $i$  calls

$TVB^i$  - The target normalized bandwidth variance of class  $i$  calls

$T_k$  - The  $k$ th time slot since the beginning of the call

$r$  - The peak rate of a connection

$VB^i$  - The normalized bandwidth variance of class  $i$  calls

$x_{ij}$  - The number of users in the local cell using bandwidth  $b_{ij}$

$x^n$  - The sequence with the length of  $n$

$y_{ij}$  - The number of users in the neighboring cells using bandwidth  $b_{ij}$

## Acknowledgments

First of all, I would like to express my sincere thanks and deep gratitude to my supervisor, Professor Victor Leung, for his constant encouragement, insightful comments, and invaluable suggestions, which benefited not only the work presented in this thesis, but also my career for a long time to come.

I would like to thank Professor Cyril Leung, Professor Hussein Alnuweiri, Professor Robert Donaldson for serving on my thesis committee. Their comments have enhanced my research in innumerable ways.

I am also very grateful to my university examiners, Professor Vikram Krishnamurthy and Professor Martin Puterman, and my external examiner, for reading and supporting my work.

During my Ph. D. program, I have had the privilege to work with Dr. Vincent Wong, who is now teaching in the ECE department of the University of British Columbia. I thank him, sincerely, for his comments on my work.

Friends and students have certainly made my studies here memorable and interesting. I would like to thank Mrs. Mary Ma, Dr. Cyril Iskander, Dr. Peter Chong, Mr. Patrick Wang, Mr. Charles Gu, Mrs. Jenny Zi, Mrs. Crystal Wang, and all the other people in the Communications Group for their friendship and enjoyable technical discussions.

I would like to express my sincere thanks to my parents for their constant love and encouragement. Many thanks to my wife, Kathleen Hu, for her love, support, understanding, and encouragement, which helped me go through this difficult yet rewarding process.

# Chapter 1 Introduction

## 1.1 QoS Provisioning in Cellular Mobile Networks

In recent years, there has been significant growth in the use of cellular mobile telephony for communications around the world. With the growing demand for integrated services supporting data traffic such as the world wide web (WWW) and e-mail as well as multimedia such as video and audio in cellular mobile networks, quality of service (QoS) provisioning is becoming more and more important. Due to the intrinsic characteristics of cellular mobile networks, such as the scarcity of wireless bandwidth, location-dependent and time-varying wireless links and user mobility, QoS provisioning in cellular mobile networks is more challenging than in wireline networks.

To utilize the radio spectrum efficiently, a cellular architecture is used in wireless mobile networks. The geographical coverage area is partitioned into *cells*, each served by a base station. Since a mobile user may change cells a number of times during the lifetime of a call, availability of wireless network resources at a call's setup time does not necessarily guarantee that wireless network resources are available throughout the lifetime of a call. Due to handoff and mobility, overload conditions can occur if the communication needs of a number of mobile terminals populating a small area exceed the total capacity of all base stations within their reach. Thus, users may experience performance degradations due to mobile handoffs. This problem will be magnified in future micro/pico-cellular networks, where handoff events may occur at a much higher rate compared to today's macro-cellular systems [63]. Call admission control (CAC) and bandwidth reservation are required to address this problem. Since forced call terminations, due to handoff blocking, are generally more objectionable than new call blocking, we consider  $P_{hd}$ , the

*probability of handoff dropping*, as a call-level QoS metric provisioned by CAC for non-adaptive traffic, which cannot change bandwidth during the lifetime of a call. As it is impractical to completely eliminate handoff call dropping, the best option is to keep  $P_{hd}$  below a target level. Moreover, maximizing resource utilization while keeping  $P_{nb}$ , the *probability of new call blocking*, below a target value is another critical factor for evaluating CAC algorithms.

Another important issue in cellular networks is mobility management (MM). Since mobile users are free to move within the area covered by the network, the network needs to first determine a particular user's location whenever there is a need to establish communication with that user. The problem of mobility management is usually divided into two parts: *paging* and *location updating*. Paging is the network operation to find the exact location of a called mobile user, whereas the location update process keeps track of each mobile's general location so as to reduce both paging cost and delay. Recent research shows that per-user mobility information plays an important role in designing efficient MM algorithms. Since per-user mobility information is very important for QoS provisioning as well, it will be helpful to consider MM and QoS provisioning jointly and have them share information with each other.

The scarce and highly fluctuating bandwidth of wireless links has motivated the development of adaptive multimedia applications that can operate over a wide range of available bandwidths. In adaptive multimedia applications, the bandwidth of a call can be dynamically adjusted to adapt to the various communication environments, especially in link

overload situations. Moreover, the systems proposed for future cellular mobile networks provide flexible radio resource allocation functions, based on resource availability conditions. Under this adaptive multimedia framework, a bandwidth adaptation (BA) algorithm needs to be used in conjunction with the CAC algorithm for QoS provisioning. CAC decides the admission or rejection of new and handoff calls, whereas BA reallocates the bandwidth of ongoing calls.

Cellular mobile networks usually have to interconnect with wireline networks for end-to-end communications. As the Internet evolves into a global communication infrastructure, Internet services and Internet communications using open standard protocols (including those being developed to support the QoS guarantees required for multimedia and premium services) are expected to play key roles in future generation cellular mobile networks. There is a growing demand for replacing the current *same-service-to-all* paradigm in the Internet with a model in which packets, applications, and users are treated differently based on their service needs. Several schemes have recently been proposed to provide QoS in the Internet, such as integrated service (IntServ), differentiated service (DiffServ) and the endpoint measurement-based admission control (EMAC) schemes. Therefore, there are two sets of different QoS definitions and mechanisms in cellular mobile networks and the Internet. However, the QoS experienced by a mobile user is end-to-end, that is, from the server to the mobile host. So, effective interconnecting schemes between cellular networks and the Internet are very important to guarantee the end-to-end QoS.

The rest of this chapter is organized as follows. Section 1.2 gives an overview of

related work. Section 1.3 discusses the motivations of this work. Section 1.4 summarizes our contributions. Section 1.5 describes the organization of this thesis.

## 1.2 Overview of Related Work

### 1.2.1 Call Admission Control for Non-Adaptive Traffic

The *guard channel policy* [41], [60] and *fractional guard channel policy* [62] determine the number of guard channels, which are reserved statically for handoffs, by considering just the status of local cell. Users are assumed to be uniformly located in any cell of the mobile network under these policies.

In a *distributed call admission control scheme* [55], not only the status of a local cell but also that of adjacent cells are considered. The total required bandwidth for both handoff and existing calls is calculated under the assumptions of exponentially-distributed channel holding time and perfect knowledge of the rate of handoff. These assumptions are unrealistic in real networks.

There have also been some research efforts to consider user mobility in designing CAC. The *shadow cluster scheme* [47] estimates future resource requirements in a collection of cells, called the shadow cluster, which a mobile is likely to visit in the future. Admission control is performed based on this estimate. However, this proposal lacks a mechanism to determine the shadow cluster in real networks, as it assumes either precise knowledge of user mobility or totally random user movement.

In [57], bandwidth is reserved in neighboring cells based on user movement predic-

tion, and an algorithm is used to control the size of reserved bandwidth pool. However, this scheme does not estimate channel holding time, and therefore, cannot be applied directly for efficient bandwidth reservation. In [24], user mobility is estimated based on the aggregate history of the handoff observed in each cell. This scheme assumes that each mobile will handoff to neighboring cells with equal probability in the mobility estimate time window. This assumption may not be realistic in general.

### 1.2.2 Mobility Management

Several MM schemes have been proposed in the literature. A good survey can be found in [78]. Two MM schemes related to this thesis are summarized as follows.

In *location area (LA)-based scheme*, the coverage area is partitioned into a number of LAs. Each LA contains a group of cells. All base stations within the same LA broadcast the identifier (ID) of its LA periodically. Each mobile terminal compares its registered LA ID with the current broadcast LA ID. A location update is triggered if the two IDs are different. Upon a call arrival for a particular mobile terminal, all the cells within its current LA are polled simultaneously, ensuring success within a single step. The LA-based update scheme is widely adopted by the current cellular systems, including the EIA/TIA IS-41 [3] and the Global System for Mobile Communication (GSM) networks [54]. The main drawback of this scheme is that for an LA with a large number of cells, a significant amount of radio bandwidth is consumed in paging for each call arrival. In addition, mobile terminals located close to an LA boundary may perform excessive location updates as they move back and forth between two LAs.

The idea of the *path-based scheme* [13] is based on a data compression algorithm proposed by Ziv and Lempel. It works as an add-on module to the underlying update scheme (e.g., LA-based), which generates the movement history. However, a real update message is not sent to the network for each symbol. As with data compression, the algorithm parses the sequence of symbols to form code words for transmissions. The network database maintains the movement history in a compact form by a *trie*, which can be considered to be a part of the user's profile. Upon a call arrival, selective paging based on the trie information is used to locate the mobile terminal.

### **1.2.3 Call Admission Control and Bandwidth Adaptation for Adaptive Traffic**

*Channel sub-rating scheme* for telephony services is proposed in [50]. "Sub-rating" refers to an occupied full-rate channel being temporarily divided into two channels at half the original rate: one to serve the existing call and the other to serve the new request. Authors in [94] consider a similar scheme with one class of non-adaptive service and one class of adaptive service. In [25], an analytical model is derived for one class of adaptive service. The extension of these schemes designed for one class of adaptive traffic to the case of multiple classes in real cellular networks may not be an easy task.

Talukdar *et al.* [71] study the trade-offs between network overload and fairness in bandwidth adaptation for multiple classes of adaptive multimedia. However, the proposed scheme in [71] does not consider maximizing wireless network utilization and may result in sub-optimal solution.

Markov decision process formulation and linear programming are used in [82]. The large state and action space problems in [82] may hinder the deployment of this scheme in practical networks. Authors in [89] use a simulated annealing algorithm to find the optimal call-mix selection. In these schemes, only the status of the local cell is considered in QoS provisioning for adaptive multimedia. It is well known that the status of neighboring cells has an increased influence on the QoS of the local cell in future multimedia cellular networks [63], and therefore, the status information of neighboring cells is very important for the QoS provisioning that can adapt to changes in the traffic pattern.

Authors in [35] and [45] make fine attempts to consider the status information of neighboring cells. However, only one class of traffic is studied and they do not consider maximizing network revenue.

#### **1.2.4 QoS Provisioning in the Internet**

The traditional approach to addressing this problem is an IntServ architecture [18], in which applications and users request a certain performance level that can be guaranteed using resource reservation and admission control mechanisms. While such architectures provide excellent QoS, they face significant deployment and scalability difficulties.

DiffServ [15] is another approach to provide QoS for the Internet. The DiffServ model is characterized by marking and creation of several traffic classes receiving differentiated treatment. DiffServ requires no per-flow admission control or signaling, and routers do not maintain any per-flow state. The combination of provisioning, service-level-agreements and DiffServ router mechanisms may prove sufficient for providing QoS for individual real-time

flows.

Attempting to combine DiffServ's superior scalability with IntServ's superior QoS, several recent papers [14], [19], [28], [42] have proposed the novel approach of using EMAC. In these designs, the end host probes the network by sending probe packets at the data rate it would like to reserve and recording the resulting level of packet losses (or congestion marks). The host then admits the flow only if the loss (or marking) percentage is below some threshold value. Since EMAC requires no explicit support from the routers, and since routers keep no per-flow state and do not process reservation requests, this is an attempt to use the regular best-effort infrastructure (with its DiffServ extensions), and by adding control algorithms at the endpoints, deliver a real-time service. Analysis and simulation results from previous work suggest that a soft real-time service based on endpoint probing may be viable.

### 1.3 Motivations

To guarantee  $P_{hd}$  below a target level and maximize resource utilization at the same time for non-adaptive traffic, several CAC schemes have been reported in the literature. Although most of these proposals have achieved this goal, these schemes are derived under unrealistic assumptions. First of all, structured cell configurations are commonly used. Circular, hexagonal or square cell configurations are often used in two-dimensional models, and a linear model is commonly used in the one-dimensional case [24], [47], [55]. Although these network topology models simplify the analyses, they do not accurately represent a real cellular network, where the cell shape and size may vary depending on the receiver sensitivity, antenna radiation pattern of the base station and propagation environment, and where the

number of neighboring cells varies from cell to cell. We believe a generalized graph model is more appropriate to represent the topology of an actual cellular network. Second, most existing schemes assume channel holding time follows an exponential distribution, which is independent and identically distributed for all calls. However, simulation studies and field data have shown that exponentially distributed channel holding time is not accurate in actual networks [31], [44]. Third, the symmetric random walk model has been quite popular among researchers in characterizing individual movement behavior. In this model, a mobile user moves to any one of the neighboring cells with equal probability after leaving a cell. This model does not take into account the direction of the mobile user. In general, a mobile user usually travels with a destination in mind. So, the mobile's location in the future is likely to be correlated with its movement history. Motivated by these observations, we investigate efficient CAC and bandwidth reservation schemes based on assumptions more realistic than existing proposals.

Another part of our work focuses on a framework combining QoS provisioning and mobility management. The motivations behind this part of our work are based on the following observations. First, it is generally assumed in the literature that the sole purpose of the location update mechanism is to aid the paging process, and that the CAC decision should be based on a different information set. However, the MM problem in cellular networks arises primarily because of user mobility. On the other hand, user mobility is also the primary reason why CAC in cellular networks is required to take extra steps to guarantee call-level QoS. So, the information used to solve one problem may be useful for solving another one. Second, recent work [13], [66], [78] has considered per-user mobility patterns to design more efficient

MM schemes. Authors in [47], [57], [83] have also considered per-user mobility patterns in CAC in cellular networks. Since analytic and simulation results in these papers show that the per-user mobility pattern can provide the basis for effective solutions that address these two system requirements, it will be helpful to consider them jointly and make them share information with each other. Third, in solving the MM problem, some schemes [8], [13], [48] only use out-of-session (i.e., between call arrivals) movements and ignore in-session (i.e., during the calls) mobility information. On the other hand, only in-session movements, but not out-of-session movements, are used in designing CAC in [24], [83]. In fact, both in-session and out-of-session movements are parts of a user's mobility pattern. Therefore, it is expected that using all available mobility information will improve the performance of both CAC and MM schemes.

In adaptive multimedia cellular mobile networks, maximizing revenue while meeting QoS constraints suggests a constrained semi-Markov decision process (SMDP) [53]. The traditional model-based solutions [61] to SMDP, which require state transition probabilities, suffer from two "curses": The *curse of dimensionality* and the *curse of modeling*. The curse of dimensionality is that the complexity and solution space in these algorithms increase exponentially as the number of states increases. QoS provisioning in adaptive multimedia cellular networks involves very large state space, which makes model-based solutions infeasible. The curse of modeling is that in order to apply model-based methods, it is first necessary to express state transition probabilities explicitly. This is very difficult in real cellular networks due to irregular network topology, different propagation environments and random user mobility. These curses suffered by model-based approaches motivate us to

pursue alternative solutions. A model-free reinforcement learning (RL)-based approach, which does not require explicit state transition probabilities, is studied in this thesis.

The motivations behind our work on CAC for cellular-to-IP internetworking are based on the experience of Transmission Control Protocol (TCP) on wireless domain. Like reliable transport protocols such as TCP, EMAC is an end-to-end scheme. It is well known that TCP is tuned to perform well over wireline links and has degraded performance over wireless links. Consequently, many modifications of TCP have been made to improve the performance of TCP in such an environment [7]. Motivated from this experience, we believe that it is better to consider the wireless environment before the deployment of EMAC than to modify it after it is widely accepted. We will study the performance of EMAC in cellular mobile networks and propose a CAC scheme for cellular-to-IP internetworking.

## 1.4 Main Contributions

The main contributions of this thesis are as follows:

- **Mobility-based predictive CAC and bandwidth reservation:** To provide the call level QoS for non-adaptive traffic in cellular mobile networks, we propose to statistically predict user mobility based on the mobility history of users. The mobility prediction scheme is derived from data compression techniques that are both theoretically optimal and good in practice. In order to utilize resources more efficiently, we predict not only the cell to which the mobile will handoff, but also when the handoff will occur. We also adaptively control the admission threshold to achieve a better balance between guaranteeing handoff dropping probability and maximizing

resource utilization. Results show that the proposed schemes meet our design goals and outperform other schemes.

- **A framework combining QoS provisioning and mobility management:** We propose a novel framework to solve QoS and mobility management problems in cellular networks using a common user mobility prediction scheme. We use all available location information of the mobile, both during calls and between calls, for a more accurate modeling of individual users' mobility. Particularly, in our framework, not only location updates between calls but also handoff records during a call lifetime are used to build the mobility trie. Results show that more effective solutions to both CAC and mobility management can be realized from this framework.
- **Solving QoS provisioning for adaptive multimedia using reinforcement learning:** We present a QoS provisioning scheme for adaptive multimedia in cellular mobile networks via reinforcement learning, which can maximize the network revenue subject to several predetermined QoS constraints. Unlike other model-based algorithms, our scheme does not require the explicit state transition probabilities, and therefore, the underlying assumptions of our scheme are more realistic than those in previous schemes. Simulation results demonstrate the superior performance of the proposed scheme to some of the existing methods.
- **A CAC scheme for cellular-to-IP internetworking:** Although EMAC has many desirable features in the wireline networks, as shown in the previous work, in this

thesis we show that several distinct characteristics in cellular mobile networks make EMAC difficult to implement. We propose a mobile-EMAC (M-EMAC) scheme to internetwork cellular networks to the Internet. Results show that the proposed scheme performs better than EMAC in cellular mobile networks.

## 1.5 Organization of the Thesis

The rest of the thesis is organized as follows. In Chapter 2, we propose and analyze the mobility-based predictive CAC and bandwidth reservation schemes. The mobility prediction approach derived from optimal data compression scheme is described. Based on the mobility prediction, efficient CAC and bandwidth reservation schemes are proposed. We present performance comparisons using simulations. In Chapter 3, we propose a framework that combines QoS provisioning and mobility management. We present numerical results and compare the performance with other schemes. In Chapter 4, we propose to solve the QoS provisioning problem for adaptive multimedia in cellular mobile networks via reinforcement learning. We describe the formulation of the problem and our reinforcement-learning-based approach. Performance comparisons with other schemes by simulations are given. In Chapter 5, we present the numerical results of EMAC performance in cellular mobile networks. A novel M-EMAC scheme is proposed and its performance is evaluated using simulations. Finally, Chapter 6 concludes the thesis with a summary of the presented work and some suggestions for future work.

## **Chapter 2 Mobility-Based Call Admission Control and Bandwidth Reservation**

### **2.1 Introduction**

In this chapter, we address the issues of CAC and bandwidth reservation by proposing and evaluating mobility-based predictive schemes under more realistic assumptions [83], [87]. In order to guarantee the handoff dropping probability, we propose to statistically predict user mobility based on the mobility history of users. Using this prediction scheme, we can predict not only the cell to which the mobile will handoff, but also when the handoff will occur. Bandwidth is reserved to guarantee some target handoff dropping probability. We also adaptively control the admission threshold to achieve a better balance between guaranteeing handoff dropping probability and maximizing resource utilization. Simulation results show that the proposed schemes meet our design goals and outperform the static reservation scheme.

The rest of this chapter is organized as follows. System models that are more realistic than those considered previously in similar work are illustrated in Section 2.2. In Section 2.3, we describe and analyze the mobility prediction schemes. Based on the mobility prediction, efficient CAC and bandwidth reservation schemes are proposed in Section 2.4. Simulation results are presented and discussed in Section 2.5. A summary is given in Section 2.6.

### **2.2 Model Description**

We consider a mobile communication network with a cellular wireless infrastructure. A handoff could fail due to insufficient bandwidth in the new cell, causing the handoff call to

be dropped. In this study, we do not consider (1) soft handoff in Code Division Multiple Access (CDMA) systems [75], in which a mobile can communicate with two or more base stations simultaneously; (2) delay-insensitive applications, which can tolerate long handoff time delay caused by insufficient bandwidth. We describe the network topology, channel holding time distribution and user mobility pattern considered in our study in the following subsections.

### 2.2.1 Network Topology

To address CAC and bandwidth reservation problems, most researchers model cellular networks by structured graphs. Circular, hexagonal or square cell configurations are often used in two-dimensional models, and a linear model is commonly used in the one-dimensional case. Although these network topologies simplify the analyses, they do not accurately represent a real cellular network, where the number of neighboring cells varies from cell to cell, and the shape and size of each cell may vary depending on receiver sensitivity, antenna radiation pattern of the base station, and the propagation environment.

Our network topology model is not restricted to structured cell configuration, such as hexagonal or linear. We use a generalized graph model to represent the actual cellular network. The network is modeled as a connected graph  $G = (V, E)$ , where the vertex-set  $V$  represents the set of base stations, each serving a single cell, and the edge-set  $E$  represents the adjacency between pairs of cells. Figure 2.1 shows an example network representation with vertex-set  $V = \{a, b, c, \dots, l\}$  and edge-set  $E = \{(a, b), (a, c), \dots, (k, l)\}$ .

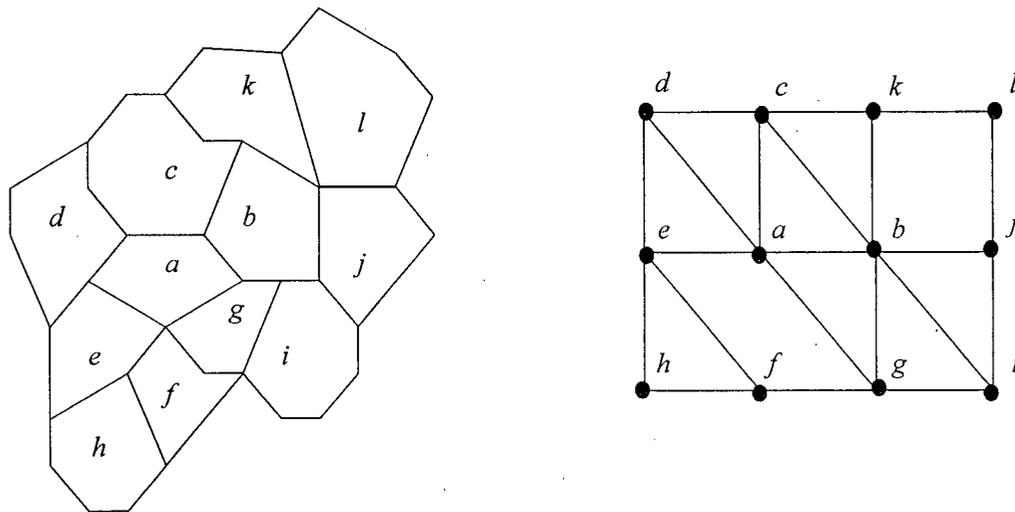


Figure 2.1 Modeling an actual cellular network

### 2.2.2 Channel Holding Time

The channel holding time is defined as the time during which a new or handoff call occupies a channel in a given cell, and it is dependent on the mobility of the user. While this is similar to the call holding time in the fixed telephone network, it is often a fraction of the total call duration in a cellular mobile network and needs not have the same statistical properties [31], [44]. Most research work on CAC and bandwidth reservation assumes the channel holding time follows an exponential distribution [41], [47], [62], which is independent and identically distributed (i.i.d.) for all cells. Like the structured models for network topology, i.i.d. exponential distribution simplifies the analyses, but does not give an accurate representation of the real characteristics of cellular networks.

We assume that the channel holding time follows a general distribution, which allows the i.i.d. exponential channel holding time assumption to be relaxed.

### 2.2.3 User Mobility Pattern

The symmetric random walk model has been quite popular among researchers in characterizing individual movement behavior [55], [62]. In such a model, a mobile user will move to any one of the neighboring cells with equal probability after leaving a cell. This model does not take into account the trajectory and channel holding time of a mobile.

In cellular mobile networks, the mobility of a user during a call can be represented by a sequence of events,  $N, H_1, H_2, H_3, \dots, H_n, \dots, E$ , where  $N$  represents the event that a new call is admitted,  $H_n$  represents the event of a mobile's  $n$ th handoff and  $E$  represents the call termination event. Note that in some cases, there are no handoff events during the lifetime of a call and thus no  $H_n$  in the sequence of events. In this sequence,  $N = (m, i, t)$ , where  $m$  represents the mobile requesting the call,  $i$  represents the original cell and  $t$  represents the time when the call arrives;  $H_n = (T_k, j)$ , where  $T_k$  is the time elapsed since the beginning of the call and  $j$  is the cell to which the mobile will handoff; and  $E = (T_k)$ . We quantize the relative time into slots of equal duration  $T$ , a design parameter. So,  $T_k$  is the  $k$ th time slot since the beginning of the call.

In general, a mobile user usually travels with a specific destination in mind. So, the mobile's location and channel holding time in the future are likely to be correlated with its movement history. Therefore, in our model, the sequence of events  $N, H_1, H_2, H_3, \dots, H_n, \dots, E$  is assumed to be generated by an  $m$ th order Markov source, in which the states correspond to the contexts of the previous  $m$  events. The probabilities of possible next events can depend on a list of  $m$  previous events.

## 2.3 Mobility Prediction

We derive probabilistic predictions of user mobility based on the accumulated behavior history of each specific mobile. The rationale behind the scheme is the observation that a user's mobility pattern is a reflection of the routines of his/her life and that most mobile users have favorite routes and habitual movement patterns. This repetitive nature of mobility patterns suggests the *stationarity* of a sequence of events generated by an  $m$ th order Markov source. Thus, we can learn those patterns from the mobility history of a particular user and predict the user's next move when those patterns reappear.

A similar prediction approach is used in [13] to solve the mobility management problem in cellular networks. This scheme records only the locations of mobile users to predict their future locations and cannot be used directly to derive efficient CAC and bandwidth reservation schemes. Although the possibility of using this method for QoS provisioning in cellular networks is mentioned in [13], as far as we know, our proposal is the first to realize this possibility. The novelty of our proposal compared to the previous one is that we record both the locations and the handoff times of mobile users. Therefore, we can derive a novel prediction method that predicts not only where a mobile user will handoff, but also when the handoff is likely to occur. Based on this novel prediction method, we further propose CAC and bandwidth reservation schemes that are more efficient than existing methods.

The prediction approach is motivated from optimal data compression methods. In data compression, a data set (e.g., a text file or an image) is decomposed into a sequence of events, and encoded using as few bits as possible. Thus, short codewords should be assigned to more

probable events and longer codewords should be assigned to less probable events. So, in order to compress data well, one has to be able to predict future data well, and hence a good data compressor should also be a good predictor. If a data compressor expects a certain character to be next with a very high probability, it will assign that character a relatively short code. If the overall code length is small, then the predictions of the data compressor must have been good.

### 2.3.1 Optimal Data Compression

In this study, we develop our mobility prediction algorithm based on the Ziv-Lempel algorithms for data compression, which are both theoretically optimal and good in practice. The original word-based Ziv-Lempel encoder [95] breaks the input string into block-to-variable codes. The algorithm parses each block of size  $n$  in a greedy manner into distinct substrings  $x_1, x_2, \dots, x_n$  with the following property: For each  $j \geq 1$ , substring  $x_j$  without its last character is equal to some previous substring  $x_i$ , where  $0 \leq i < j$ . Substring  $x_j$  is encoded by the value  $i$ , using  $\lceil \log_2(j-1) \rceil$  bits, followed by the ASCII encoding of the last character of  $x_j$ , using  $\lceil \log_2 \alpha \rceil$  bits, where  $\alpha$  is the size of the input sequence's alphabet. Because of this *Prefix Property*, substrings parsed so far can be efficiently maintained in a trie [46].

The equivalent character-based Ziv-Lempel algorithm builds in on-line fashion a probabilistic model (or a trie) that feeds probability information to an arithmetic coder [80], which encodes a sequence of probability of  $p$  using  $\log_2(1/p) = -\log_2 p$  bits. We show by example how these algorithms work.

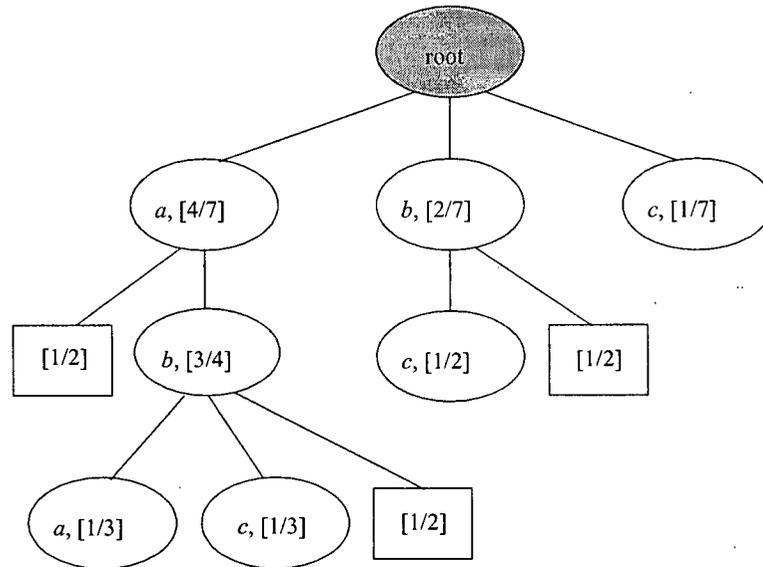


Figure 2.2 The trie constructed in Example 2.1

**Example 2.1:** Let the alphabet be  $\{a, b, c\}$ . We consider an input string “*ababcab-cababcab ...*” that the Ziv-Lempel encoder parses as “*(a)(b)(ab)(c)(abc)(aba)(bc)(ab...)*...” Each substring in the parse is encoded as a pointer followed by a character. In particular, the match “*ab*” of the sixth substring “*aba*” is encoded using  $\lceil \log_2 5 \rceil$  bits with a value 3, since the match “*ab*” is the third substring, and the last character “*a*” is encoded using  $\lceil \log_2 3 \rceil$  bits, since the alphabet size is 3.

In the character-based version of Ziv-Lempel encoder, a trie is built when each previous substring ends. The trie at the end of the seventh substring is pictured in Figure 2.2. There are four previous substrings beginning with an “*a*”, two beginning with a “*b*” and one beginning with a “*c*.” The character “*a*” is therefore assigned a probability of  $\frac{4}{7}$  at the root, “*b*” is

assigned a probability of  $\frac{2}{7}$  at the root, and “c” is assigned a probability of  $\frac{1}{7}$  at the root. Similarly, of the four substrings that begin with an “a,” three begin with an “ab,” giving the probability of  $\frac{3}{4}$  for “b,” and so on. Any sequence that leads from the root of the trie to a leaf traverses a sequence of probabilities of  $P_1, P_2, P_3, \dots$ , whose product  $\prod_i P_i$  equals  $\frac{1}{7}$ . The arithmetic coder encodes the sequence with  $\left\lceil -\log_2 \prod_i P_i \right\rceil = \lceil \log_2 7 \rceil = 3$  bits. Note that the square nodes in Figure 2.2 denote the last nodes ending the sequences.

### 2.3.2 Mobility Prediction

Our mobility prediction scheme is based on the character-based version of the Ziv-Lempel algorithm. The sequence of events  $N, H_1, H_2, H_3, \dots, H_n, \dots, E$  during the lifetime of a call corresponds to the substring in the Ziv-Lempel algorithm. The mobility database of every mobile at specific time holds a *mobility trie*, which is a probability model corresponding to that of the Ziv-Lempel algorithm. Each node, except for the root, in the mobility trie preserves the relevant statistics that can be used to predict the probability of following events. As in data compression, the mobility trie of the mobile is built in an on-line fashion. When a mobile requests a new call, the predictor sets the current node to the root of the trie according to the mobile, cell and time, and calculates the probabilities of all possible events of this mobile. Upon recording an actual event of the mobile, the predictor “walks” down the trie and is ready for the next prediction. When an event is not in the mobility trie, a prediction fault is gener-

```

initialize mobility trie :=null
initialize number_of_event :=0
loop
  wait for next event e
  if (e is a new call)
    look for a trie with a root of e in the database
    if (cannot find such a trie)
      create a trie :=single node (the root)
    endif
  else
    if (e exists in the trie)
      number_of_event :=number_of_event + 1
    else
      create a leaf e
    calculate the probabilities of possible events
    based on the number_of_event of leaves
  forever

```

Figure 2.3 A pseudo-code of mobility prediction

ated and the trie is updated. A pseudo-code description of the mobility prediction scheme is given in Figure 2.3.

Figure 2.4 shows an example of the mobility trie of mobile  $m$  at cell  $a$  in the time interval 8:00-9:00 a.m. When the mobile requests a new call in cell  $a$  in the time interval 8:00-9:00 a.m., we can use the statistics preserved in the nodes of its mobility trie to predict the probabilities of the next possible events of this mobile: it will terminate the call without handoffs in the 2nd time slot with probability of  $\frac{2}{56}$ , handoff to cell  $b$  in the 2nd time slot with probability of  $\frac{15}{56}$ , and so on.

### 2.3.3 Analysis of the Mobility Prediction Scheme

We first analyze the optimality of the word-based Ziv-Lempel algorithm and show that the character-based Ziv-Lempel algorithm is as least as good as the word-based approach.

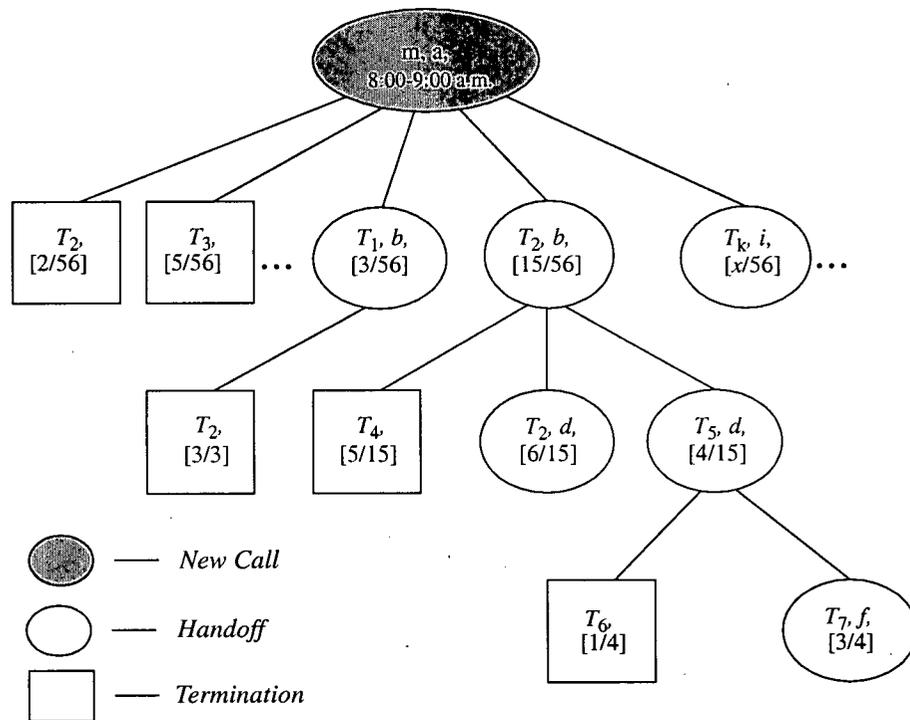


Figure 2.4 A mobility trie used for mobility prediction

Then, we will establish that our mobility prediction scheme inherits the optimality of these data compression algorithms.

Given a sequence  $x^n$  of length  $n$  over an alphabet  $A$  of  $\alpha$  letters and an *information lossless* (IL) compressor  $C$  accepting inputs over  $A$ , let  $|C(x^n)|$  denote the length, in bits, of the output that  $C$  produces on  $x^n$ . The compression ratio  $\rho_C(x^n)$  attained by  $C$  for  $x^n$  is [95]:

$$\rho_C(x^n) = \frac{|C(x^n)|}{n \log_2(\alpha)}. \quad (2.1)$$

Define  $\rho_\sigma(x^n)$  as the best compression ratio attainable for  $x^n$  by any IL compressor of  $\sigma$  states. Sequence  $x^n$  is parsed into different phrases:  $x^n = x_1, x_2, \dots, x_t$ , and  $t(x^n)$  is the maximal possible number of distinct phrases. Define:

$$q(x^n) = \frac{t(x^n) \log_2(t(x^n))}{n \log_2(\alpha)}. \quad (2.2)$$

A result in [95] shows that

$$\rho_\sigma(x^n) \geq q(x^n) - \delta(\sigma, n) \text{ with } \lim_{n \rightarrow \infty} \delta(\sigma, n) = 0. \quad (2.3)$$

So,  $q(x^n)$  is a lower bound on the compression ratio attainable for  $x^n$  by any codebook.

The Ziv-Lempel incremental parsing algorithm achieves for any sequence  $x^n$  given to it a compression ratio that is (asymptotically) equal to  $q(x^n)$ , and thus the algorithm is universal and asymptotically optimal [67].

In [9], it has been shown that the code length obtained in the character-based version of the Ziv-Lempel algorithm is as least as good as that obtained using the word-based approach. Hence, the optimality result in [95] holds without change for the character-based approach.

We define the *event fault rate* to be the total number of event faults that our mobility prediction algorithm incurs, divided by the total number of events. Also, we define the

*expected fault rate* to be the best possible fault rate achievable by any prediction algorithm that makes its prediction based only on the past history.

A result in [76] shows this:

*If the source is a stationary  $m$ th order Markov source, the expected event fault rate of our prediction algorithm is within an additive factor of  $O(1/\sqrt{n})$  from the expected event fault rate of the source, where  $n$  is the length of the event sequence.*

From these, we see that our mobility prediction algorithm inherits the asymptotic optimality of the Ziv-Lempel algorithm. By modeling the sequence of events during the lifetime of a call as that generated by a stationary  $m$ th order Markov source and predicting next events using the mobility prediction scheme derived from the Ziv-Lempel algorithm, we can predict not only to which cell a mobile will handoff, but also when the handoff will occur.

### **2.3.4 Implementation Considerations**

The mobility prediction scheme proposed above maintains the statistics in a trie, which can be stored in the user profile in cellular mobile networks. An important issue is how this model can be implemented. In fact, a trie is a multiway tree with a path from the root to a unique node for each string represented in the tree. There are many ways to implement the nodes of a trie. The simplest approach is to create an array of pointers for each node in the trie with a pointer for each character of the input alphabet (Figure 2.5-a). This method can waste considerable memory space, particularly if some characters of the alphabet are rarely used. An alternative is to use a linked list at each node, with one item for each possible branch

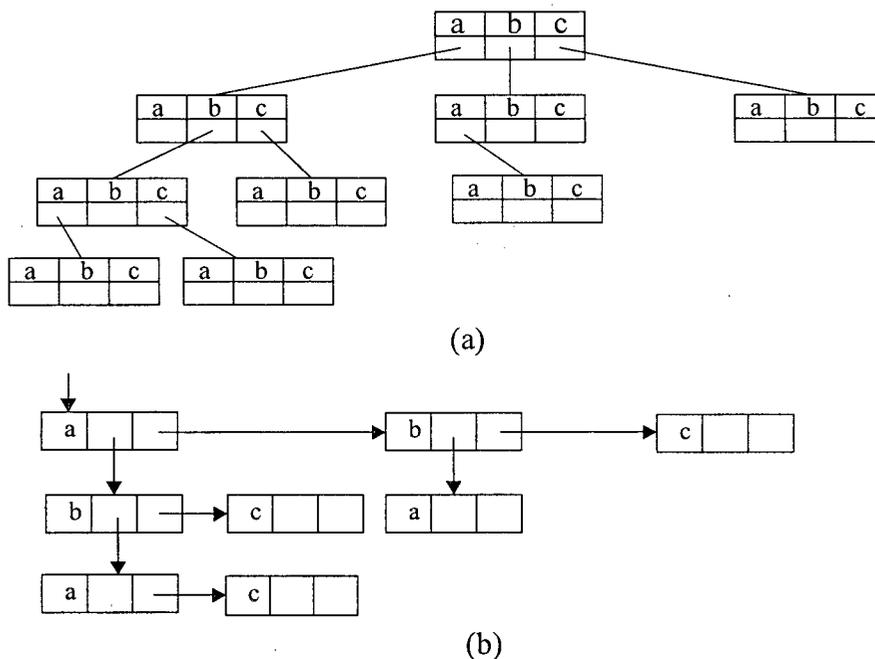


Figure 2.5 Implementation of trie nodes in Figure 2.2: (a) array and (b) linked list.

(Figure 2.5-b). The space needed for a node in the linked list requires two pointers, one counter and a symbol. A straightforward implementation of this would consume 13 bytes, and these could be packed into perhaps 11 bytes. This uses memory economically, but can be more processing intensive. Some improvement may be achieved by moving an item to the front of the list each time it is used. A trie can also be implemented as a single hash table with an entry for each node. The memory consumed by a trie can be reduced by truncating it prematurely at a shallow depth, and using some other data structure for subsequent characters. For further details, the reader can consult books on algorithms and data structures.

In practice, in order to reduce the memory and computation complexity, it is desirable to limit the size of the data structure for prediction. Several techniques are known for limiting data structure size in [69]. An explicit upper bound  $S$  is placed on the size of the data

structure. The data structure is either frozen when its size reaches  $S$ , flushed and rebuilt when its size reaches  $S$ , or frozen when its size reaches  $S/2$  and a new one is built while the old one is used for prediction. There are also more sophisticated techniques that use tags to determine what nodes are to be deleted when its size reaches  $S$  [21]. In the simulations, a linked list with 11 bytes at each node is used and the scheme in [21] is employed to maintain the size  $S = 40$ .

## 2.4 Call Admission Control and Bandwidth Reservation

### 2.4.1 Calculation of $P(i, j, T_k)$

Our approach is based on the predicted mobility of each user. We calculate  $P(i, j, T_k)$ , the probability that a mobile originally in cell  $i$  will visit cell  $j$  during time slot  $T_k$ . From the mobility trie, we can see that a mobile taking different paths can visit a certain cell in the same slot. Using the *total probability theorem* [58], we must add all of these probabilities to get  $P(i, j, T_k)$ . Example 2.2 shows how to get this probability.

**Example 2.2:** A mobile  $m$  requests a new call at cell  $a$  in the time interval 8:00-9:00 a.m. From the mobility trie in Figure 2.4, we can see that  $m$  can take several different paths to visit cell  $b$ . We describe these paths by sequences of events:

Path 1:  $N(m, a, 8:00-9:00 \text{ a.m.}), H(T_1, b), E(T_2)$ .

By Path 1,  $m$  will visit cell  $b$  in  $T_1$  and  $T_2$  with probability:  $\frac{3}{56} \times \frac{3}{3} = \frac{3}{56}$ .

Path 2:  $N(m, a, 8:00-9:00 \text{ a.m.}), H(T_2, b), E(T_4)$ .

By Path 2,  $m$  will visit cell  $b$  in  $T_2$ ,  $T_3$  and  $T_4$  with probability:  $\frac{15}{56} \times \frac{5}{15} = \frac{5}{56}$ .

Path 3:  $N(m, a, 8:00-9:00 \text{ a.m.}), H(T_2, b), H(T_2, d)$ ...

By Path 3,  $m$  will visit cell  $b$  in  $T_2$  with probability:  $\frac{15}{56} \times \frac{6}{15} = \frac{6}{56}$ .

Path 4:  $N(m, a, 8:00-9:00 \text{ a.m.}), H(T_2, b), H(T_5, d)$ ...

By Path 4,  $m$  will visit cell  $b$  in  $T_2$ ,  $T_3$ ,  $T_4$  and  $T_5$  with probability:

$$\frac{15}{56} \times \frac{4}{15} = \frac{4}{56}.$$

So,

$$P(a, b, T_1) = \frac{3}{56},$$

$$P(a, b, T_2) = \frac{3}{56} + \frac{5}{56} + \frac{6}{56} + \frac{4}{56} = \frac{18}{56},$$

$$P(a, b, T_3) = \frac{5}{56} + \frac{4}{56} = \frac{9}{56},$$

$$P(a, b, T_4) = \frac{5}{56} + \frac{4}{56} = \frac{9}{56} \text{ and}$$

$$P(a, b, T_5) = \frac{4}{56}.$$

### 2.4.2 The Most Likely Cell-Time

When a mobile is active in cell  $i$ , we can get the most likely cell-time (MLCT) of that

mobile, a cluster of time units at a cluster of cells when and where a mobile will most likely visit in the future. We select cell  $j$  and time slot  $T_k$  with  $P(i, j, T_k)$  greater than zero to form the MLCT of this mobile.

### 2.4.3 Bandwidth Reservation

Using  $P(i, j, T_k)$ , the probabilities of handing off from cell  $i$  into cell  $j$  during time slot  $T_k$  of mobile  $m$ , we can obtain the required bandwidth  $B_r(i, j, m, T_k)$  to be reserved in cell  $j$  during the time slot  $T_k$  for the expected handoff of mobile  $m$  from cell  $i$ :

$$B_r(i, j, m, T_k) = P(i, j, T_k) \cdot B(m) \quad , \quad (2.4)$$

where  $B(m)$  is the bandwidth required by  $m$ . In this study, we assume that the traffic characteristics and the desired packet-level QoS guarantees (e.g., delay, jitter, loss and throughput) can together be represented by *effective bandwidth*. Techniques for computing the effective bandwidth for different traffic characteristics and QoS requirements can be found in [29], [39]. Moreover, the reserved bandwidth  $B_r(j, T_k)$ , which is the aggregate bandwidth to be reserved in cell  $j$  during  $T_k$ , is calculated as:

$$B_r(j, T_k) = \sum_{m \in M, i \in I} B_r(i, j, m, T_k) \quad , \quad (2.5)$$

where  $M$  is the set of mobiles which will handoff to cell  $j$  from a set of cells  $I$  during  $T_k$ . Finally, the free bandwidth left after the reservation is:

$$B_f(j, T_k) = B - B_r(j, T_k) \quad , \quad (2.6)$$

where  $B$  is the total bandwidth in cell  $j$ .

#### 2.4.4 Call Admission Control and Bandwidth Reservation for New Calls

When a new call arriving at mobile  $m$  with a bandwidth requirement  $B(m)$  requires admission to cell  $i$ , the CAC algorithm first checks if the current free bandwidth of cell  $i$  can support the call. The call is rejected if the cell does not have enough free bandwidth. Otherwise, CAC will check the availability of free bandwidth in the MLCT of this mobile. The checking result can be written as:

$$Check(j, T_k, B(m)) = \begin{cases} 1, & B_j(j, T_k) \geq B(m) \\ \frac{B_j(j, T_k)}{B(m)}, & \text{otherwise} \end{cases} \quad (2.7)$$

Based on these values, the new call will be admitted if the following holds:

$$\sum_{j, T_k \in MLCT} P(i, j, T_k) \cdot Check(j, T_k, B(m)) \geq \alpha \sum_{j, T_k \in MLCT} P(i, j, T_k), \quad (2.8)$$

where  $\alpha$  is the *admission threshold* and should be controlled adaptively. We will describe how to control this threshold in the next subsection.

When a new call is admitted, bandwidth is reserved in the mobile's MLCT. And the free bandwidth in the MLCT is updated accordingly.

#### 2.4.5 Adaptive Control of Admission Threshold

The mobility prediction functions may not work well for some mobile users, especially those who do not have favorite routes. Moreover, if the admission threshold  $\alpha$  is

too small, the handoff dropping probability may exceed the target value; if  $\alpha$  is too large, resource utilization will decrease. So, admission threshold  $\alpha$  should be controlled adaptively.

We calculate  $P_{hd}$ , the handoff dropping probability of a mobile, by dividing the number of handoff drops to the total number of its calls recorded in the mobility trie. Let  $TP_{hd}$  denote the target value of handoff dropping probability of mobile  $m$ . If  $P_{hd} < TP_{hd}$ , admission threshold  $\alpha$  is decreased by  $\varepsilon$ , a design parameter; otherwise,  $\alpha$  is increased by  $\varepsilon$ . The calculation of  $P_{hd}$  and the update of admission threshold are done upon call completion.

By adaptive control of  $\alpha$ , we can achieve a better balance of guaranteeing  $P_{hd}$  and maximizing resource utilization.

#### 2.4.6 Call Admission Control and Bandwidth Reservation for Handoff Calls

When a mobile  $m$ , with bandwidth requirement  $B(m)$ , requires handoff to cell  $i$ , CAC will calculate  $P(i, j, T_k)$  and get the MLCT of  $m$  based on the mobility trie. Bandwidth is reserved for  $m$  in its MLCT accordingly. The CAC algorithm will admit the handoff call if the current free bandwidth of cell  $i$  can support the call.

### 2.5 Simulation Results and Discussions

In this section, we present and discuss the simulation results of the proposed schemes as well as comparisons with two other CAC schemes.

We consider a coverage area that consists of 40 base stations, each of which has 6

neighbors on average. The distance between two base stations is 1 mile. Since most mobile users have favorite routes, we assume that each mobile user has 5 possible different paths in the network. The user will take these 5 paths with probability of 0.5, 0.2, 0.1, 0.1, 0.1 respectively. Among the cells within a path, mobile users can have a new call request with equal probabilities. During a call, the mobile will stay at the original cell or move along the path. If a call does not terminate when the mobile reaches the end of the path, it will stay at the end cell of that path. The path is generated as follows: (1) Select two nodes in the graph randomly as original and destination nodes; and (2) Whenever the mobile user leaves the current cell, it moves to a neighboring cell that is closest to the destination. Note that two paths with at least one edge not in common are different paths, and that different mobile users can have the same paths.

Also, we apply the following assumptions in our model:

1. Each cell has a fixed link capacity of 40 bandwidth units (BUs).
2. Time is quantized into units of  $T = 30$ s.
3. A call is either for voice (requiring 1 BU) or video (requiring 4 BUs).
4. Call durations are the same for all calls and exponentially distributed with a mean value of 3 minutes.
5. Call requests are generated according to a Poisson process with rate  $\lambda$  (calls/cell/m).
6. The speeds of mobiles are uniformly distributed between 0 and 40 miles/hour.
7. The target hand off dropping probabilities are the same for all mobiles:  $P_{hd} = 0.01$ .
8. Admission threshold  $\alpha$  is initialized to 1 in each simulation and adaptive factor  $\varepsilon = 0.02$ .

The offered load is calculated as follows:

$$\text{Offered Load} = 3 \cdot \lambda \cdot ((1 - P_v) \cdot 4 + P_v),$$

where  $P_v$  is the percentage of voice in the offered traffic. The physical meaning of the offered load is the total bandwidth units required on average to support all existing calls.

Simulations start without any pre-memorized information of mobiles. Long-term handoff dropping probability, new call blocking probability and utilization are obtained for a 100h simulation time duration. During each simulation, a mobility trie is constructed for each mobile and its mobility is predicted. Based on the prediction, an MLCT is constructed. Then CAC algorithm will check the availability of bandwidth and decide to admit or reject the new call and handoff call requests using the algorithms described in Section 2.4. If the call is admitted, bandwidth is reserved in the mobile's MLCT accordingly.

Figure 2.6 shows  $P_{nb}$  and  $P_{hd}$  as functions of the offered load for two values of  $P_v$ : 0.8 and 1. The probabilities of handoff dropping are kept below the target values 0.01 irrespective of the offered load and  $P_v$ . This shows that the proposed CAC and bandwidth reservation schemes achieve one of our goals: keeping  $P_{hd}$  below a target level. We also observe that  $P_{nb}$  and  $P_{hd}$  increase as  $P_v$  decreases under the same offered load. This is because the video calls need more bandwidth. Figure 2.7 shows the average utilization as a function of the offered load with different values of  $P_v$ .

Since the time slot is used in our mobility prediction scheme, the selection of time slot duration  $T$  will influence both the convergence speed and the network utilization. We study

this issue in the following. In simulations, we choose 3 values of  $T$ , 15s, 30s and 90s. Figure 2.8 shows the handoff dropping probabilities as functions of simulation time with different time slot durations. In the beginning of the simulation, the system knows little about the mobility of mobile users and the handoff dropping probabilities cannot be kept below the target value 0.01. As the simulation goes on, the prediction tends to converge and the target handoff dropping probability can be guaranteed. From Figure 2.8 we can see that the convergence is faster when  $T$  is 90s than when  $T$  is 30s, which is in turn faster than when  $T$  is 15s. The reason for this is that the mobility sequence will be long when  $T$  is small, and when  $T$  is large, the sequence will be short. The longer the sequence, the slower the convergence speed. It seems that it is better to have a large value of  $T$ , since the convergence will be fast. However, a large value of  $T$  means that the prediction is not accurate, and results in low utilization, which can be seen clearly in Figure 2.9. The utilization is higher when  $T$  is 15s than when  $T$  is 30s, which is higher than when  $T$  is 90s. Therefore, choosing a suitable value of  $T$  according to real network conditions is very important for getting the best performance from the proposed scheme.

We also compare the proposed CAC and bandwidth reservation schemes with the well known static reservation scheme [41], [60]. In the static reservation scheme, a set of bandwidth is reserved statically for handoff calls. In our simulation, we consider 4 BUs and 5 BUs reserved permanently for handoff calls in each cell. For comparison, we call our scheme cell-time-reservation.

Figures 2.10 and 2.11 show that the static reservation scheme with 4 and 5 BUs

reserved for handoff calls can keep  $P_{hd}$  below the target value of 0.01 when the network has a light load, but the reserved bandwidth is not enough when the offered load becomes heavier. Hence, this scheme cannot achieve the design goal. Although the static reservation scheme has almost the same  $P_{hd}$  compared with our scheme when the network load is lighter, its  $P_{nb}$  is higher in this area, i.e., it admits less new calls than our scheme for any given  $P_{nb}$ . In the static reservation scheme,  $P_{hd}$  may be kept below the target value by statically reserving more bandwidth for handoff calls. However, this will result in higher  $P_{nb}$ , which means lower utilization if  $P_{nb}$  were to be reduced to an acceptable level. From these results, we can see that our proposed cell-time-reservation scheme achieves a better balance of guaranteeing  $P_{hd}$  and maximizing utilization.

## 2.6 Summary

In this chapter, we have proposed call admission control and bandwidth reservation schemes for non-adaptive traffic in cellular mobile networks based on assumptions more realistic than existing proposals. The proposed schemes are applicable to arbitrary cell topologies and the channel holding time can follow a general distribution. The sequences of events of new call admission, handoffs and call termination are modeled by stationary  $m$ th order Markov sources. We derived novel probabilistic predictions of next events based on the mobility history of users, using an algorithm motivated by optimal data compression. Based on the mobility prediction of where and when the mobile will handoff to the next cell, CAC and bandwidth reservation schemes have been developed. The performance of the proposed schemes have been studied using computer simulations. Results show that our schemes can achieve a better balance of guaranteeing handoff dropping probability while maximizing

resource utilization, and they outperform the static reservation scheme.

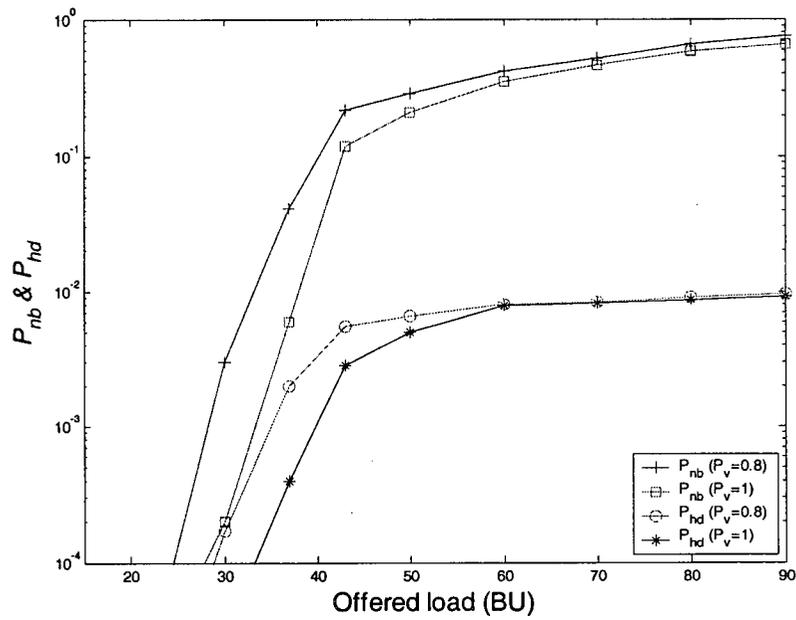


Figure 2.6  $P_{nb}$  and  $P_{hd}$  vs. offered load

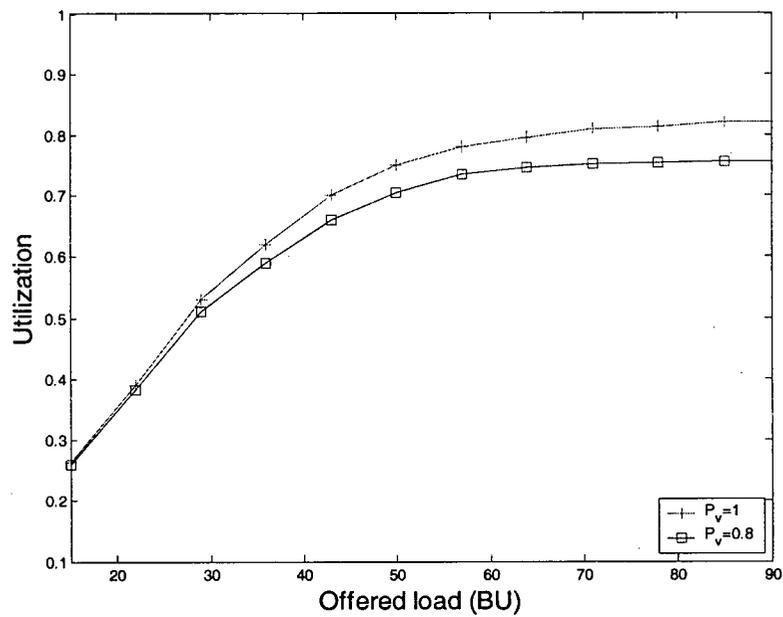


Figure 2.7 Utilization vs. offered load

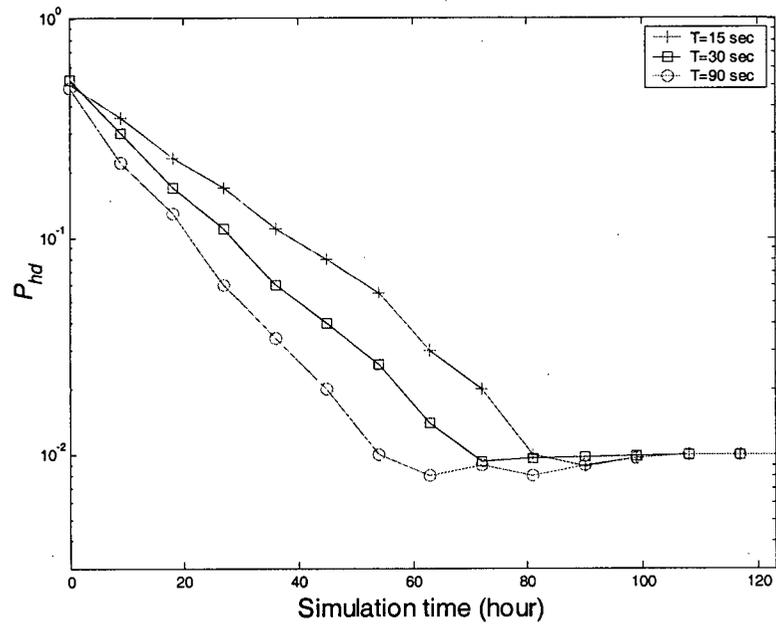


Figure 2.8  $P_{hd}$  vs. simulation time with different values of time slot duration

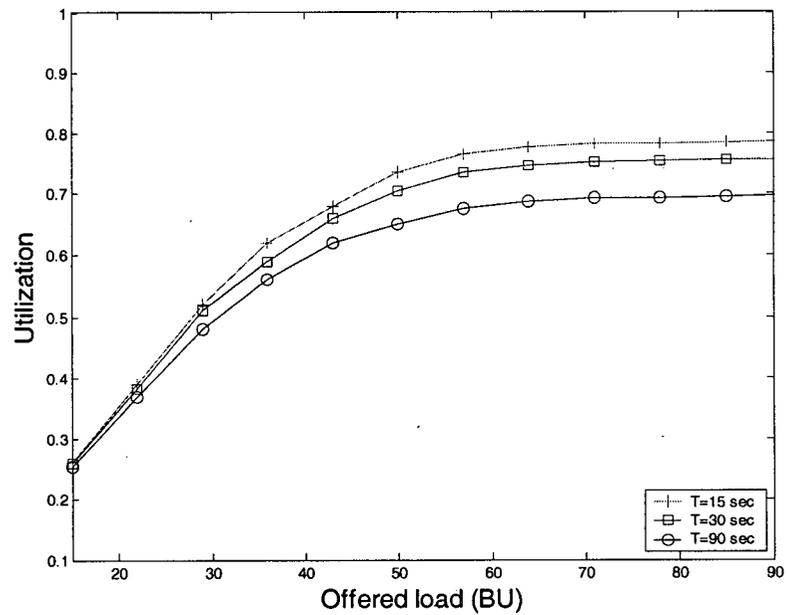


Figure 2.9 Utilization vs. offered load with different values of time slot duration

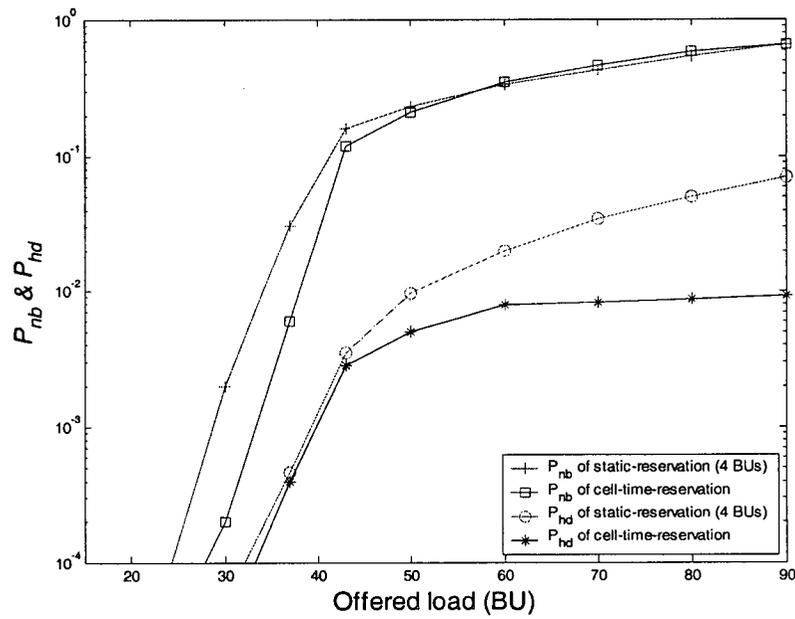


Figure 2.10 Comparison with static reservation (4 BUs)

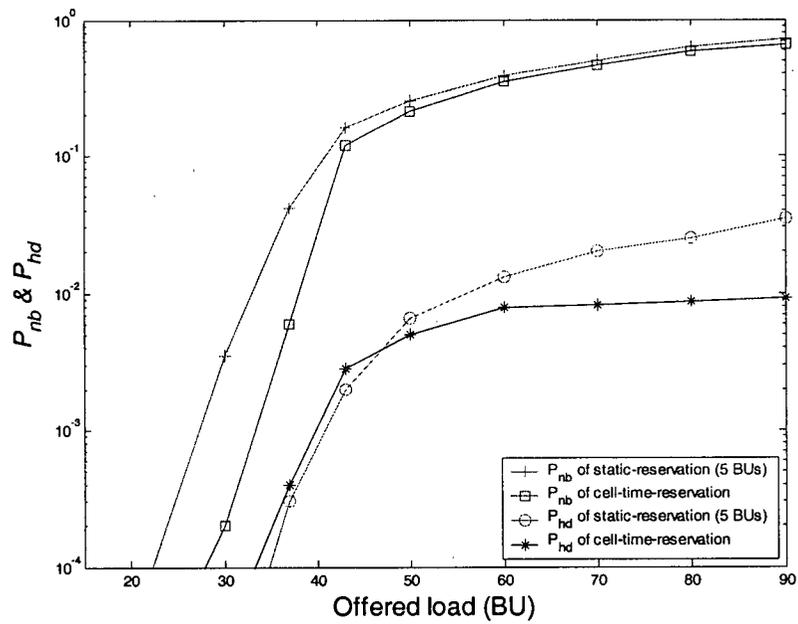


Figure 2.11 Comparison with static reservation (5 BUs)

## **Chapter 3 A Framework Combining QoS Provisioning and Mobility Management**

### **3.1 Introduction**

In this chapter, we propose a framework combining QoS provisioning and mobility management in cellular networks [84], [88], [89]. Since user mobility is the primary reason why both CAC and mobility management problems arise, the information used to solve one problem may be useful for solving another one. Moreover, recent work shows that the per-user mobility pattern plays an important role in both of these two issues. It will be helpful to consider them jointly and have them share information with each other. Therefore, we can use the mobility prediction scheme in Chapter 2 as the key component in this framework, which can be used in both locating/paging mobiles and making admission decisions. Using this framework, we propose a new path-based MM scheme and a new CAC scheme that use all available mobility information. Simulation results are presented to show the performance improvements of the new QoS and MM schemes in this framework.

The rest of this chapter is organized as follows. The common mobility prediction scheme is described in Section 3.2. In Section 3.3, we describe the location update and paging schemes in our framework. The new CAC scheme is described in Section 3.4. A summary of this chapter is given in Section 3.5.

### **3.2 Common Mobility Prediction Scheme**

If the position of a user can always be predicted in advance, then no explicit update is necessary and the optimal location area is the one that minimizes paging cost, i.e., a single

cell. On the other hand, if the network knows accurately where the mobile will move when the mobile needs to perform a handoff during a call, a more efficient CAC scheme can be designed. Therefore, the key component in our framework combining QoS provisioning and MM together is a common mobility prediction scheme that can be used for both paging mobiles and making admission decisions. In Chapter 2, we show that the mobility prediction scheme derived from the optimal data compression algorithm is optimal in terms of event fault rate. However, in Chapter 2, we only use the in-session (i.e., during calls) mobility information and ignore the out-of-session (i.e., between call arrivals) movements. In fact, both in-session and out-of-session movements are part of a user's mobility pattern. It is expected that using all available mobility information will improve the performance of both CAC and MM schemes.

Although the Ziv-Lempel-based prediction scheme, which is used in Chapter 2, is shown to be optimal, the *prediction by partial match* (PPM) scheme usually outperforms the Ziv-Lempel algorithm due to implementation considerations and a faster convergence rate [9]. The basis of the PPM algorithm of order  $m$  is a set of  $m+1$  Markov predictors. A Markov predictor of order  $j$  predicts the next event based on the  $j$  immediately preceding events. In order for PPM to work well, the network needs to maintain all contexts of order  $0, 1, \dots, m$ . Note that although an enhanced symbol-wise model is used in [13], it cannot store all the contexts of different orders, which will eventually affect the convergence rate and prediction performance. Here, we use another trie [9] which will combine all contexts together into a single data structure. The following example demonstrates the data structure and how PPM works.

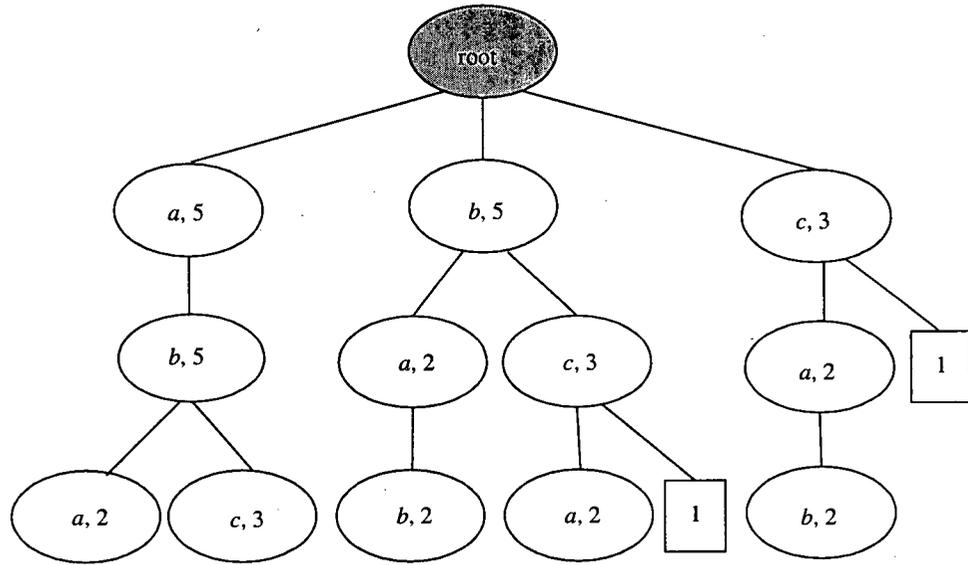


Figure 3.1 The trie constructed in Example 3.1

**Example 3.1:** Assume that a mobile has visited a sequence of cells, “*ababcabcababcab...*,” which is same as the input string in Example 2.1. The mobile can send the “compressed” sequence of cells to the network. A trie at the end of the 7th subsequence is built at the network side, shown in Figure 3.1. Assume that the predictor in the network knows the last three cells visited by the mobile are “*abc*” and wants to predict the probabilities of cells that the mobile will next visit. We can estimate the probability distribution with order 0, 1, and 2. First, we can see that only one child “*a*” follows path “*bc*” in an order-2 prediction. So, we assign  $P_{2a} = 1$ ,  $P_{2b} = 0$ , and  $P_{2c} = 0$ . Second, in an order-1 prediction based on the context “*c*”, we assign  $P_{1a} = 1$ ,  $P_{1b} = 0$  and  $P_{1c} = 0$ , since one child “*a*” is in the subtrees rooted at node “*c*”. Finally, in an order-0 estimate, we start from the root of the whole trie and get  $P_{0a} = 5/13$ ,  $P_{0b} = 5/13$  and  $P_{0c} = 3/13$ . Given a blending weight vector  $w = [w_0, w_1, w_2]$ , the blended probability assignment is:

$$P_a = w_2 \cdot P_{2a} + w_1 \cdot P_{1a} + w_0 \cdot P_{0a}$$

$$P_b = w_2 \cdot P_{2b} + w_1 \cdot P_{1b} + w_0 \cdot P_{0b}$$

$$P_c = w_2 \cdot P_{2c} + w_1 \cdot P_{1c} + w_0 \cdot P_{0c}$$

The weights can be fixed values or adapt as prediction proceeds to give more emphasis to high-order models. Different ways of choosing the weights correspond to different PPM methods [9].

### 3.3 Mobility Management in the Combined Framework

In this section, we will present the MM scheme in the proposed combined framework using both out-of-session and in-session location information. Since analytic and simulation results in previous work show that path-based MM scheme is an efficient approach, we adopt the path-based scheme proposed in [13]. However, unlike with the original path-based scheme, in which the sole purpose of the location update is to aid paging, and where paging process depends only on the information provided by a location update, we use all available location information of a mobile for a more accurate modeling of individual users' mobility patterns. We present the original path-based scheme, our new path-based scheme, and numerical results of performance comparisons between the two schemes in the following subsections.

#### 3.3.1 The Original Path-Based Scheme

In path-based MM scheme [13], movement path history rather than current location is sent in an update message. The movement path history consists of a list of cell IDs the mobile terminal has crossed after the last update. The path-based scheme works as an add-on module

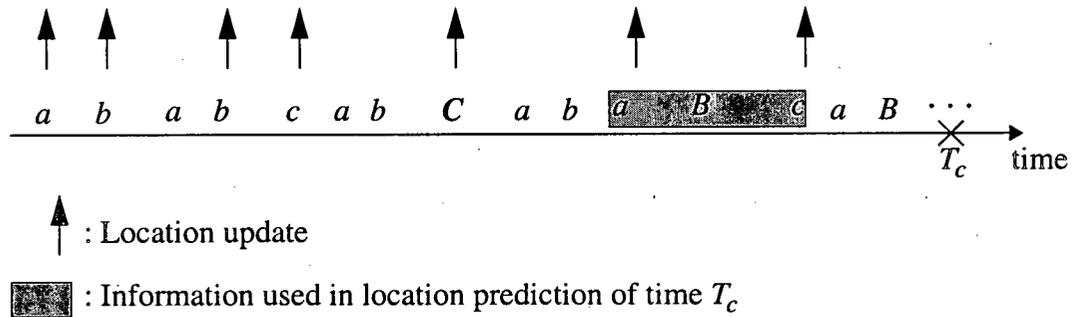


Figure 3.2 The original path-based scheme

to the underlying update scheme (e.g., LA-based), which generates the movement path history. However, a real update message is not sent to the network by the mobile for each movement. As with data compression, the algorithm sends the “compressed” movements to the network. For simplicity, assume that a mobile has visited a sequence of cells, “*ababcabCaBabcaB...*,” which is same as that shown in Example 3.1. The capital letters represent the cells in which the mobile had a call, i.e., the in-session location information, and other letters represent the cells in which the mobile did not have a call, i.e., out-of-session location information. The transmissions of update messages in the original path-based scheme are shown in Figure 3.2, where  $T_c$  is the current time.

Assume that the mobile is called at time  $T_c$ . The network can predict the location probability from previous update messages using the mobility prediction scheme in Section 3.2. After getting probabilities  $P_a$ ,  $P_b$  and  $P_c$ , the network will page the mobile by ordering cells according to a decreasing sequence of probability values.

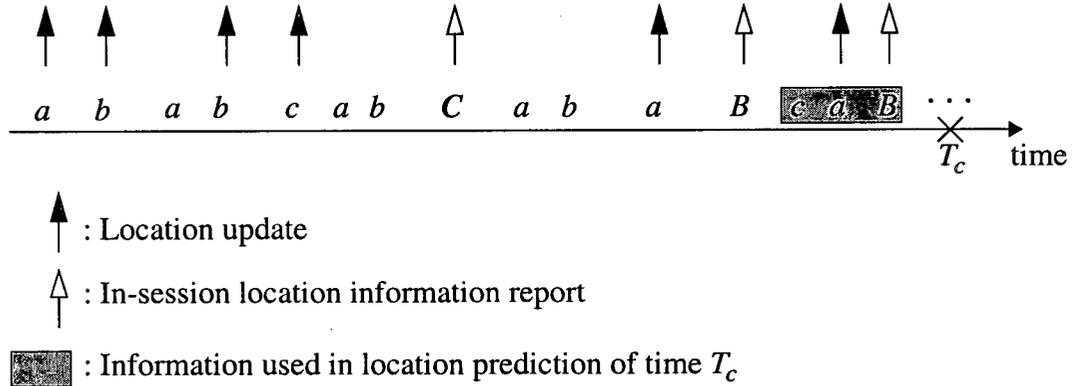


Figure 3.3 The proposed path-based scheme

### 3.3.2 The New Path-Based Scheme

The original path-based scheme does not use the in-session location information. However, the network knows the exact location of a mobile during the call session, which is very useful in searching the mobile, especially when the call-to-mobility ratio is high. Therefore, we propose to use this information in the new path-based MM scheme. In the proposed new path-based scheme shown in Figure 3.3, a mobile will give the network all unreported location information during the call session. Note that this in-session location information report requires little extra resources, because a mobile must report its location and any location changes to the network during each call in current cellular systems. Otherwise, the call cannot be delivered correctly. After the in-session location information report, the mobile will send a location update message only after it transverses a new path unseen before. For example, in Figure 3.3, a new out-of-session sequence “*aba*” is sent to the network after the in-session location information report in “*C*.” There are 6 location updates before the time  $T_c$  in Figure 3.3, which is less than 7 location updates in the original path-based scheme. Without loss of generality, since some location information of a mobile is

reported to the network during the call session, the location sequences reported by out-of-session location updates in the new scheme should be shorter than those in the original scheme. Therefore, fewer update messages are needed in the new scheme, especially when the call-to-mobility ratio is high.

On the network side, the same mobility trie is constructed to predict user mobility. However, in the new scheme, we can use more up-to-date information to make the prediction. For example, “*caB*” is used in Figure 3.3 and “*aBc*” is used in Figure 3.2. Obviously, prediction in the new scheme will be more accurate. If there is no in-session information between the last update and the current time, the prediction will be the same in the new scheme as in the original one.

### 3.3.3 Simulation Results and Discussions

The simulation environment is similar to that in Chapter 2. We assume that the cell residence time follows an i.i.d. Gamma distribution with average time  $1/u_r$  minutes. New calls arrive according to Poisson process with rate  $\lambda$  per minute and call durations are exponentially distributed with a mean value of  $1/u_d$ , which is 3 minutes.  $\rho = \lambda/u_r$  is the call-to-mobility ratio. The underlying location update scheme is movement-based [8], which generates the movement path history. In this scheme, each mobile terminal counts the number of boundary crossings between cells incurred by its movements. A location update is performed when this number exceeds a predefined movement threshold. We use 1 as the threshold in the simulations.

Given specific parameters, we can get the update messages when using the original

path-based scheme, denoted as “Update (original).” The term “Update (new)” denotes the number of update messages when using the new path-based scheme. The performance gain is the ratio Update (original)/Update (new). A similar procedure is used in evaluating the paging process. “Paging (original/new)” denotes the number of cells paged in the original/new scheme and Paging (original)/Paging (new) is the performance gain. Figures 3.4 and 3.5 show the performance gain versus the call-to-mobility ratio with different cell residence time. From these two figures, we can see that the performance gain is always greater than one. That is,  $\text{Update (original)} \geq \text{Update (new)}$  and  $\text{Paging (original)} \geq \text{Paging (new)}$ . This implies that the proposed new scheme has less cost for mobility management, and therefore, has better performance than the original one. When the call-to-mobility ratio is small, the performance gain is not significant in these two figures. However, when the call-to-mobility ratio is large, the new scheme needs much fewer update messages and paging cells than the original one. This is because the new scheme uses both in-session and out-of-session information in the location management process. The higher the call-to-mobility ratio, the more in-session mobility information is used in the new scheme. We also observe that cell residence time has some effects on the performance gain in these figures. When cell residence time is small (i.e.,  $u_r$  is large), the mobile travels a lot and has more mobility information. The new scheme can give a better model and more accurate prediction of user mobility, and therefore, has more performance gain over the original one.

### 3.4 Call Admission Control in the Combined Framework

Due to in-session user mobility, the CAC scheme needs to perform mobility-related QoS provisioning in cellular networks. If the in-session user mobility can be predicted,

efficient CAC schemes can be designed. In this section, we propose a novel CAC scheme within our combined framework. The key idea in our CAC scheme is to predict where and when the mobile will subsequently visit when the mobile originates or accepts a call. Then, we can check the availability of resources in these cells when resources are needed. If the resources are available, they are reserved for this mobile to guarantee the target  $P_{hd}$ , the probability of handoff call being dropped. In our framework, we can easily predict to which cell the mobile will move using our common mobility prediction scheme described in Section 3.3. However, since we do not record the time in our combined framework, we need to determine the time interval of the visit to make efficient resource reservations. Fortunately, we can use the mobile station positioning technology developed and standardized recently in [64] to get this time interval. We present these ideas in the following subsections.

### 3.4.1 Time Interval Prediction

For the purpose of satisfying the US FCC E-911 location requirement and for driving location information applications in Europe, recent advances in mobile station positioning technology can locate a mobile within a certain accuracy level (i.e., 50m in 67% of cases and 150m in 95% of cases for handset based location solution, mandated by US FCC) [64]. We can use this location information to improve the system performance and increase wireless system functionality for location-based commercial services. In this study we are interested in how to design better CAC schemes using this technology. A variety of basic technologies are available to determine accurate position locations. One of them is the enhanced observed time difference (E-OTD) technology [3], which has been standardized for location services in GSM-based systems.

In the E-OTD method, the unknown mobile position  $\mathbf{p} = (x, y)$  is estimated by processing the time difference of arrival (TDOA) measurements between the mobile and at least three base transceiver stations (BTSs) of known co-ordinates,  $\text{BTS}_1, \text{BTS}_2, \dots, \text{BTS}_N$ . The TDOA between  $\text{BTS}_1$  (the serving BTS) and  $\text{BTS}_i$  ( $i = 2, \dots, N$  - the neighbor BTSs) defines a hyperbola whose foci coincide with the co-ordinates of the two BTSs. At least two hyperbolas (i.e., two TDOAs) are minimally sufficient to estimate the mobile position, as shown in Figure 3.6. The TDOA is defined as the geometric time difference (GTD).  $\text{GTD}_i = \tau_1 - \tau_i$  ( $i = 2, \dots, N$ ), where  $\tau_1$  and  $\tau_i$  are the absolute propagation delays from the serving and the  $i$ th neighbor BTSs to the mobile. In principle, GTD can be determined by measuring the difference in reception epochs (REs) of bursts synchronously transmitted by the BTSs and received by the mobile. In practice, due to the non-synchronized BTSs, the different transmission epochs must be taken into account.  $\tau_i$  is defined as  $\tau_i = t_{Rx_i} - t_{Tx_i}$ , where  $t_{Rx_i}$  and  $t_{Tx_i}$  are, respectively, the reception and transmission epochs of the burst from the  $i$ th BTS. With these definitions,  $\text{GTD}_i$  can be written as the difference between the observed time difference (OTD) and the real time difference (RTD):

$$\text{GTD}_i = \tau_1 - \tau_i = (t_{Rx_1} - t_{Tx_1}) - (t_{Rx_i} - t_{Tx_i}) = \text{OTD}_i - \text{RTD}_i.$$

The mobile itself assists the location estimation by measuring the OTDs. The RTDs are measured by network monitoring equipment deployed in fixed and known locations. The positioning problem in the absence of measurement errors can be formulated with a set of  $N-1$  equations describing hyperbolas with foci at the BTSs' co-ordinates  $(x_i, y_i)$  ( $i = 1, \dots, N$ ) and intersecting at  $\mathbf{p} = (x, y)$ :

$$c \cdot \text{GTD}_i = \sqrt{(x-x_1)^2 + (y-y_1)^2} - \sqrt{(x-x_i)^2 + (y-y_i)^2}, i = 1, \dots, N \quad (3.1)$$

where  $c$  is the speed of light. Exact non-iterative solutions of this problem can be found in [30].

If we can get the estimations of a mobile's locations, it is not difficult for us to estimate the velocity of the mobile, from which we can predict the time interval of the visit. However, in real applications, equation 3.1 is rendered inconsistent by measurement noise. A linear regression setup can be used to smooth the data for more accurate velocity and position estimation of a mobile [40]. In this scheme,  $k$  previous estimations are used to get the mobile's current estimated velocity and position. Let  $\hat{\mathbf{p}}(t_n) = [\hat{x}(t_n), \hat{y}(t_n)]$ ,  $n = 1, \dots, M$  represent the estimated locations at subsequent time points  $t_n$ . The velocity of a mobile can be obtained [40]:

$$\hat{v}(t_n) = \|\hat{\mathbf{v}}(t_n)\| = [\hat{v}_x^2(t_n) + \hat{v}_y^2(t_n)]^{1/2}, \quad (3.2)$$

$$\text{where } \hat{v}_x(t_n) = \frac{\sum_{j=n-k+1}^n (t_j - i)[\hat{x}(t_j) - \bar{x}]}{\sum_{i=n-k+1}^n (t_j - i)} \quad \text{and} \quad \hat{v}_y(t_n) = \frac{\sum_{j=n-k+1}^n (t_j - i)[\hat{y}(t_j) - \bar{y}]}{\sum_{i=n-k+1}^n (t_j - i)}$$

with  $i = \left( \sum_{j=n-k+1}^n t_j \right) / k$ ,  $\bar{x} = \left( \sum_{j=n-k+1}^n \hat{x}(t_j) \right) / k$  and  $\bar{y} = \left( \sum_{j=n-k+1}^n \hat{y}(t_j) \right) / k$ . The estimated position of the mobile at time  $t_n$  is:

$$\hat{\mathbf{p}}(t_n) = t_n \cdot \hat{\mathbf{v}}(t_n) + \hat{\mathbf{o}}(t_n), \quad (3.3)$$

where  $\hat{\mathbf{o}}(t_n)$  is the estimated original position at time  $t_0 = 0$ .  $\hat{o}_x(t_n) = \bar{x} - \hat{v}_x(t_n) \cdot i$  and  $\hat{o}_y(t_n) = \bar{y} - \hat{v}_y(t_n) \cdot i$ .

After getting the estimated velocity and the position of a mobile from equations 3.2 and 3.3, we can predict the time interval during which a mobile will visit a cell. Let  $t_a(i, j)$  denote the time when the mobile in cell  $i$  will arrive at cell  $j$ , and  $t_d(i, j)$  denote the time when the mobile in cell  $i$  will depart from cell  $j$ .  $t_a(i, j)$  and  $t_d(i, j)$  can be calculated as:

$$t_a(i, j) = \frac{d(\mathbf{p}(t), j)}{v(t)} \quad \text{and} \quad t_d(i, j) = t_a(i, j) + \frac{d(j)}{v(t)}, \quad (3.4)$$

where  $d(\mathbf{p}(t), j)$  is the distance between the current position  $\mathbf{p}(t)$  and the boundary of cell  $i$  and  $j$ , and  $d(j)$  is the route distance inside cell  $j$ . We assume this distance information is available from an internal map of the relevant area, since such information is essential to provide some location services in the future.

### 3.4.2 Call Admission Control Scheme

The CAC scheme in our combined framework is similar to that in Chapter 2. We calculate  $P(i, j, t_a, t_d)$ , the probability that a mobile original in cell  $i$  will visit cell  $j$  during the time interval  $t_a$  and  $t_d$ . Assume that the call durations follow exponential distribution with a mean value of  $1/u_d$ .

$$\begin{aligned} P(i, j, t_a, t_d) &= P(i, j) \cdot P(\text{The call is not terminated by } t_d) \\ &= P(i, j) \cdot \int_{t_a}^{\infty} u_d e^{-u_d t} dt \\ &= P(i, j) \cdot e^{-u_d t_d}, \end{aligned} \quad (3.5)$$

where  $P(i, j)$  can be calculated using the common mobility prediction scheme in Section 3.2. When a mobile is active in cell  $i$ , we can get the most likely cell-time (MLCT) of that mobile, a cluster of cells and time where and when the mobile will most likely visit in the future. We

select cell  $j$  and time  $t_a, t_d$  with  $P(i, j, t_a, t_d)$  greater than zero to form the MLCT of this mobile.

Using  $P(i, j, t_a, t_d)$ , we can obtain the required bandwidth  $B_r(i, j, m, t_a, t_d)$  to be reserved in cell  $j$  for the expected handoff of  $m$  from cell  $i$ :

$$B_r(i, j, m, t_a, t_d) = P(i, j, t_a, t_d) \cdot B(m) , \quad (3.6)$$

where  $B(m)$  is the effective bandwidth required by  $m$ . Moreover, the reserved bandwidth  $B_r(j, t_a, t_d)$ , which is the aggregate bandwidth to be reserved in cell  $j$  during the time interval  $t_a$  and  $t_d$ , is calculated as:

$$B_r(j, t_a, t_d) = \sum_{m \in M, i \in I} B_r(i, j, m, t_a, t_d) , \quad (3.7)$$

where  $M$  is a set of mobiles which will visit cell  $j$  from a set of cells  $I$  during the time interval.  $B_r(j, t_a, t_d)$  may not be a constant value due to the continuous time and the summation of equation 3.7. Finally, the free bandwidth left after the reservation is:

$$B_f(j, t_a, t_d) = B - B_r(j, t_a, t_d) , \quad (3.8)$$

where  $B$  is the total bandwidth in cell  $j$ . Again, note that  $B_f(j, t_a, t_d)$  may not be a constant value. Let  $\min[B_f(j, t_a, t_d)]$  denote the minimum value of free bandwidth in cell  $j$  during the time interval.

When a new call arriving at mobile  $m$  with a bandwidth requirement  $B(m)$  requires admission to cell  $i$ , the CAC algorithm first checks if the current free bandwidth of cell  $i$  can support the call. The call is rejected if the cell does not have enough free bandwidth.

Otherwise, CAC will check the availability of free bandwidth in the MLCT of this mobile.

The checking result can be written as the following:

$$Check(j, t_a, t_d, B(m)) = \begin{cases} 1, & \min[B_f(j, t_a, t_d)] \leq B(m) \\ \frac{\min[B_f(j, t_a, t_d)]}{B(m)}, & \text{otherwise} \end{cases} \quad (3.9)$$

Based on these values, the new call will be admitted if the following holds:

$$\sum_{j, t_a, t_d \in MLCT} P(i, j, t_a, t_d) Check(j, t_a, t_d, B(m)) \geq \alpha \sum_{j, t_a, t_d \in MLCT} P(i, j, t_a, t_d) \quad , \quad (3.10)$$

where  $\alpha$  is the admission threshold and should be controlled adaptively.

### 3.4.3 Simulation Results and Discussions

The simulation environment is similar to that in Chapter 2. Some differences are as follows. We use 100 BUs as the link capacity to test our schemes in large link capacity situations. Moreover, we assume that the position information of a mobile is available but with error. The error of mobile position estimation follows a normal distribution  $N(0, \sigma^2)$  with  $\sigma = 51$  m, which will have an accuracy level of 50 m in 67% of the cases [3]. From the mobile's position information, we predict the time interval during which a mobile will visit a cell.

Figure 3.7 shows the new call blocking probability  $P_{nb}$  and handoff dropping probability  $P_{hd}$  as functions of offered load with two values of voice ratio. We observe that the handoff dropping probabilities are kept below the target value of 1%, which is similar to that in Chapter 2. Next, we evaluate the impact of different values of the adaptive factor  $\epsilon$ ,

which is used to adaptively control the admission threshold to achieve a better balance of guaranteeing  $P_{hd}$  and maximizing resource utilization. Figure 3.8 shows  $P_{hd}$  as functions of simulation time in a cell with the offered load of 100 and  $u_r = 0.2$ . We experiment with three values of  $\epsilon$ , 0.01, 0.02 and 0.04. In the beginning of the simulation, the system has little information about users and cannot predict mobility accurately. Then, the admission threshold  $\alpha$  is increased by a step of  $\epsilon$  to keep  $P_{hd}$  below the target value. From Figure 3.8, we observe that the scheme under-reacts when  $\epsilon = 0.01$  and cannot quickly keep  $P_{hd}$  below 1%, whereas it over-reacts when  $\epsilon = 0.04$  and makes the  $P_{hd}$  fluctuate near the target value. The value of 0.02 can keep  $P_{hd}$  below the target value with reasonable speed and has no fluctuation. Choosing a suitable value of  $\epsilon$  according to real network conditions is very important to get the best performance from the proposed scheme. We use  $\epsilon = 0.02$  in other simulations.

We also compare the proposed CAC with two other schemes: (1) OKS98 [57]; and (2) YL01 [83] presented in Chapter 2. In OKS 98, bandwidth is reserved in all neighboring cells when a mobile has a new call or handoffs to a new cell. We choose the best scheme in [57] called *movement-based and bandwidth-based* for comparisons. In this scheme, different bandwidth is reserved in different neighboring cells based on user movement prediction and an algorithm is used to control the size of reserved bandwidth pool. However, OKS98 does not address how to predict user mobility. We can input the mobility prediction in our framework to OKS98. Thus, it can use the information regarding where the mobile will move in the reservation. Note that OKS98 does not use the prediction of when the mobile will move. In YL01, the mobility prediction scheme and CAC scheme are similar to those considered here. However, only in-session, but not out-of-session, mobility information is

collected and used for prediction. The proposed scheme is called combined framework (CF)-CAC in the following.

Figure 3.9 shows the new call blocking probability and handoff dropping probability of CF-CAC, OKS98 and YL01. We observe that OKS98 can also keep  $P_{hd}$  below the target value. However, it has higher  $P_{nb}$ , which means fewer new calls being admitted. This is because the proposed scheme predicts not only to which cell the mobile will handoff but also when the handoff will occur. Based on these mobility predictions, we can reserve bandwidth more efficiently. Because both YL01 and CF-CAC predict where and when of handoffs, the  $P_{nb}$  and  $P_{hd}$  of YL01 are quite similar to those of CF-CAC, which are omitted in Figure 3.9. The utilization comparisons of these three schemes with different offered loads are shown in Figure 3.10. As expected, CF-CAC and YL01 have similar utilization, which is higher than that in OKS98. Figure 3.11 plots the  $P_{hd}$  as functions of simulation time in a cell in YL01 and CF-CAC when the offered load is 100 and  $u_r = 0.2$ . From this figure, we can observe that  $P_{hd}$  can be kept below the target value in CF-CAC much faster than that in YL01. This is because although both schemes record user mobility history to make predictions, the proposed scheme uses all mobility information, both in-session and out-of-session, which makes the predictions converge faster compared with that of YL01. The proposed scheme is more desirable in a real network where user mobility and traffic patterns change over time, since it can adapt to changes faster, and therefore, make more accurate predictions.

### 3.5 Summary

In this chapter, we have presented a novel framework combining QoS provisioning

and MM using all available mobility information. The key component of this framework is a common mobility prediction scheme, which can be used in both paging mobiles and making admission decisions. In addition, a new path-based MM scheme in the combined framework was proposed. Numerical results show that the new scheme offers performance gains over the original in terms of reduced update and paging costs. Finally, a CAC scheme based on our framework has been proposed. The proposed CAC scheme can predict where the mobile will handoff using the common mobility prediction scheme, and when the handoff will occur using positioning technology. Simulation results show that the proposed scheme meets our design goal and outperforms the CAC scheme in Chapter 2 and an existing scheme in [57].

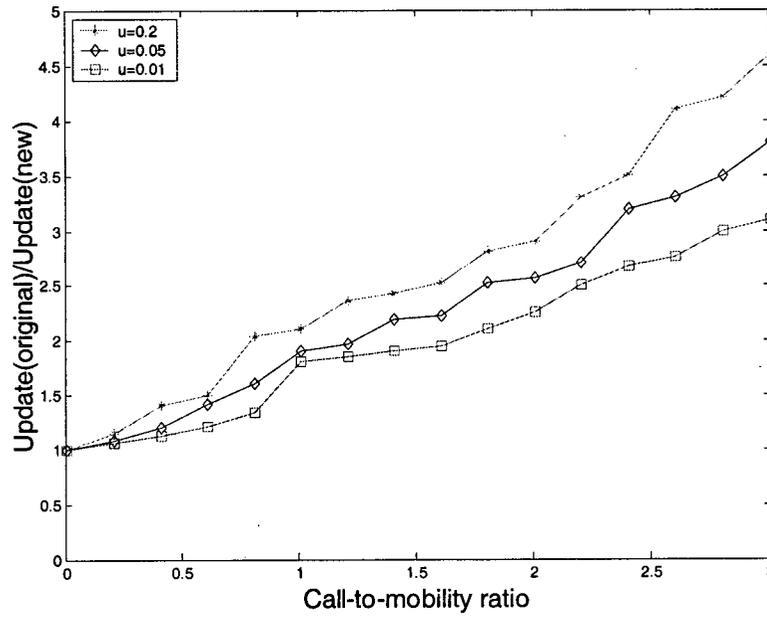


Figure 3.4 Location update performance gain vs. call-to-mobility ratio

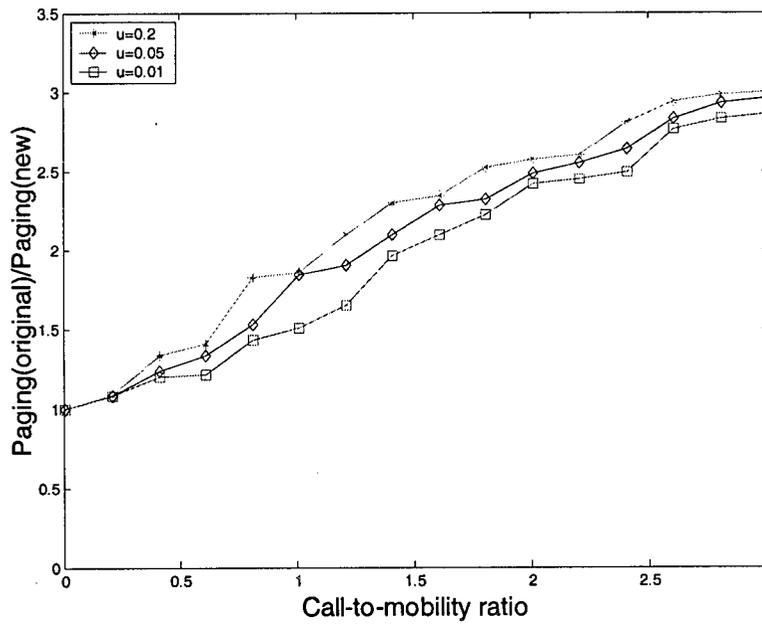


Figure 3.5 Paging performance gain vs. call-to-mobility ratio

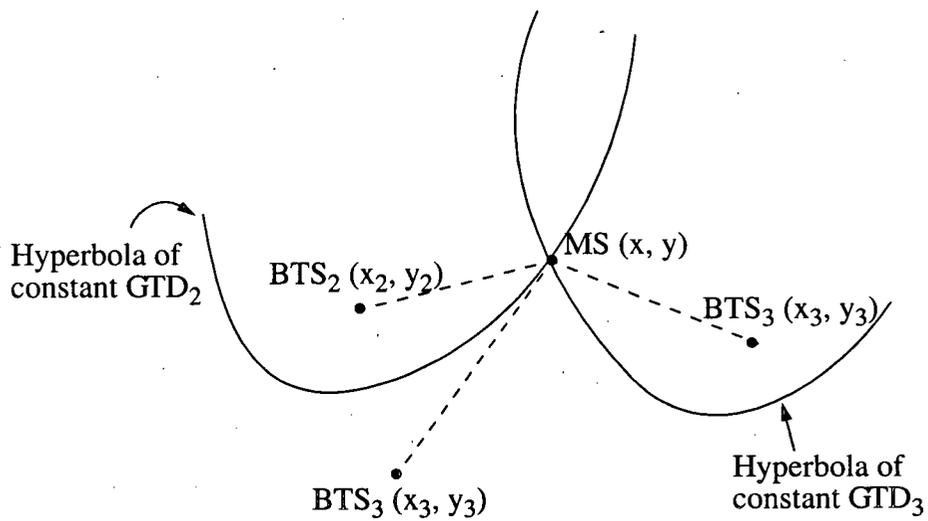


Figure 3.6 Hyperbolic positioning in E-OTD,  $N = 3$

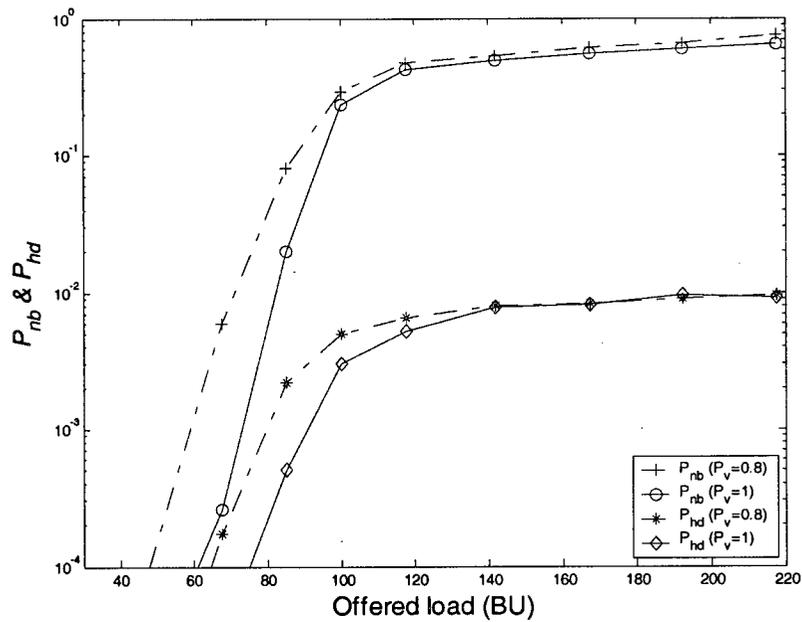


Figure 3.7  $P_{nb}$  and  $P_{hd}$  vs. offered load

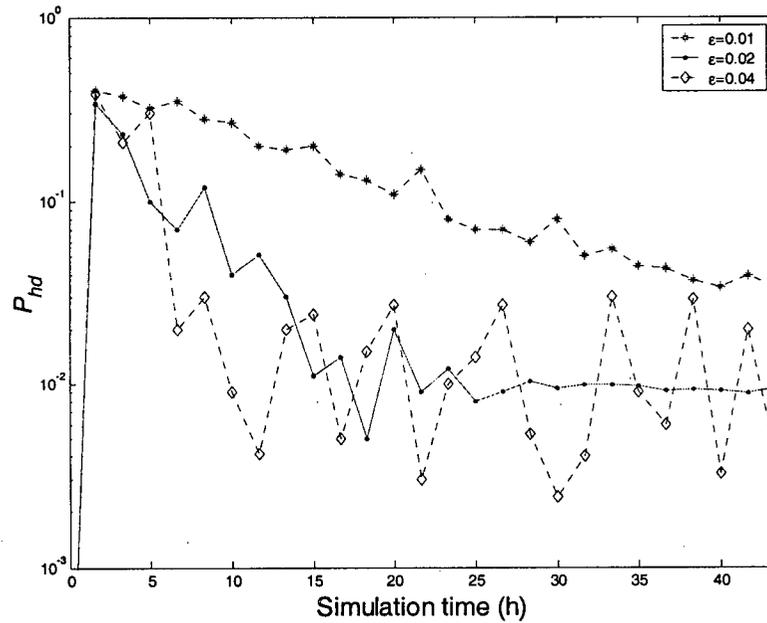


Figure 3.8  $P_{hd}$  vs. simulation time with different values of adaptive factor

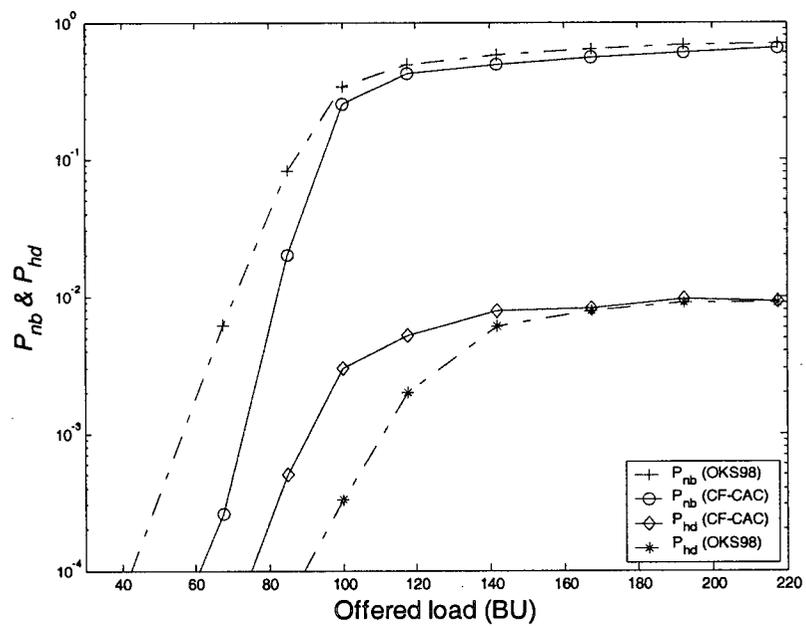


Figure 3.9 Comparison with YL01 and one of OKS98 schemes

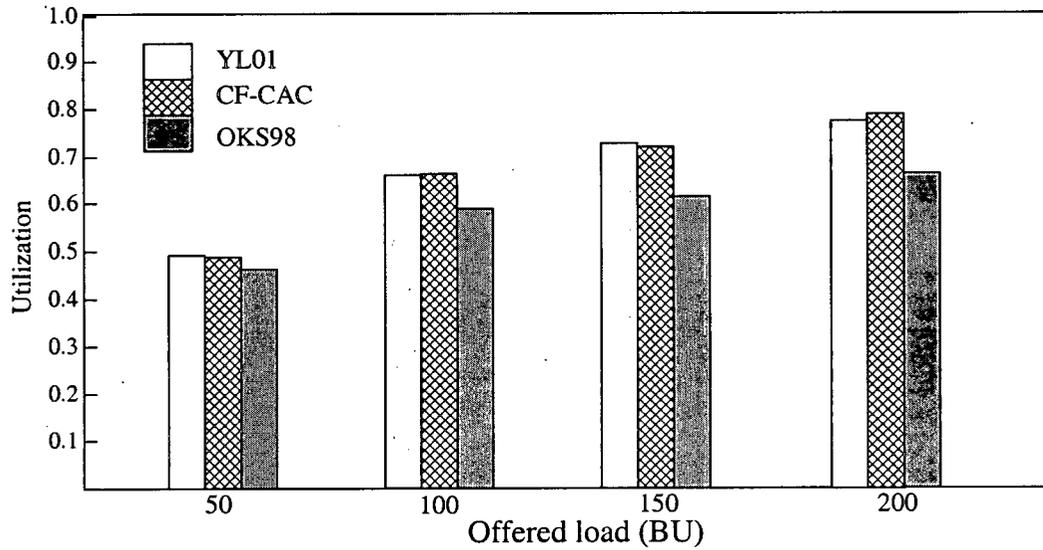
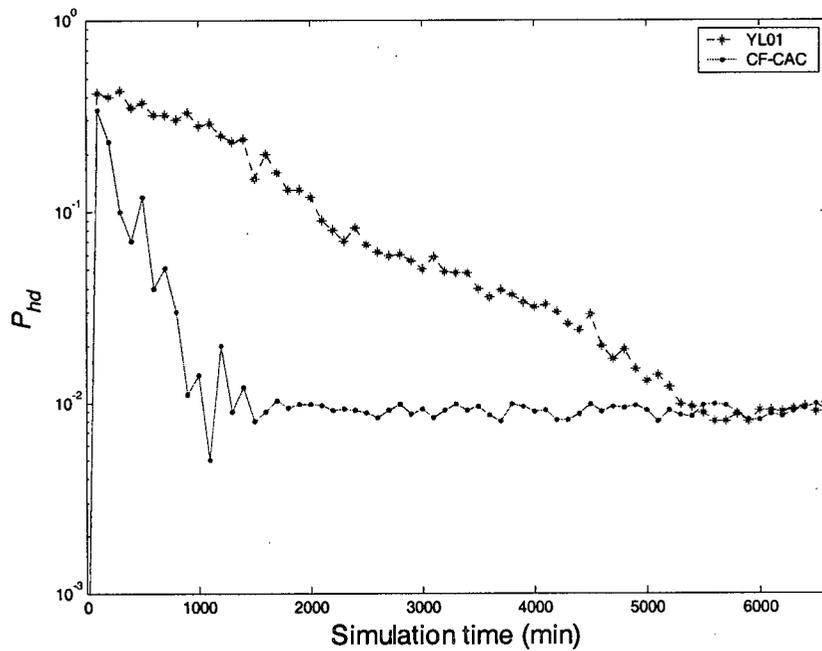


Figure 3.10 Comparison with YL01 and one of OKS98 schemes



## Chapter 4 QoS Provisioning for Adaptive Multimedia

### 4.1 Introduction

The last two chapters only consider non-adaptive traffic and non-adaptive cellular mobile networks, where the bandwidth of new or ongoing calls cannot be changed. However, in recent years, the scarce and highly fluctuating bandwidth of wireless links has motivated the development of adaptive multimedia applications that can operate over a wide range of available bandwidths. In adaptive multimedia applications, the bandwidth of a call can be dynamically adjusted to adapt to the various communication environments, especially in the link overload situations. Under this adaptive multimedia framework in cellular mobile networks, a bandwidth adaptation algorithm needs to be used in conjunction with the CAC algorithm for QoS provisioning.

This chapter presents a novel average reward reinforcement learning approach to solve the QoS provisioning problem for adaptive multimedia in cellular networks that aims to maximize the network revenue while satisfying several predetermined QoS constraints [90], [91]. The novelties of the proposed scheme are as follows:

1. The proposed scheme takes into account the effects of the status of neighboring cells with multiple classes of traffic, enabling it to dynamically adapt to changes in the traffic condition.
2. The underlying assumptions of the proposed scheme are more realistic than those in previous schemes. Particularly, the scheme does not need prior knowledge of system state transition probabilities, which are very difficult to estimate in practice due to the irregular

network topology, different propagation environment and random user mobility.

3. The algorithm can control the adaptation frequency effectively by formulating the cost of the bandwidth adaptation in the model. It is observed in [25], [71] that frequent bandwidth switching among different levels may consume a lot of resources and may be even worse than a large degradation ratio. The proposed scheme can control the adaptation frequency more effectively than previous schemes.
4. Trading off action space with state space is proposed in our scheme. As mentioned in [82], the large action space problem may hinder the deployment of the scheme in [82] in real systems. With the approach of trading off action space with state space, the large action space problem in QoS provisioning can be solved.
5. Average reward RL is used in this study. Average reward RL is more suitable than discounted reward RL in solving QoS provisioning problems in adaptive multimedia cellular networks.
6. *Handoff dropping probability, average allocated bandwidth and inter-class fairness* are considered simultaneously as QoS constraints in our scheme and can be kept below the predetermined values.

The rest of this chapter is organized as follows. Section 4.2 describes the QoS provisioning problems in the adaptive multimedia framework. Section 4.3 introduces the average reward reinforcement learning algorithm. Our new approach to solve the QoS provisioning is presented in Section 4.4. Section 4.5 discusses some implementation issues of our approach. Section 4.6 presents and discusses the simulation results. Finally, we conclude this chapter in Section 4.7.

## 4.2 QoS Provisioning in Adaptive Framework

### 4.2.1 Adaptive Multimedia Applications

Originally, adaptive multimedia applications are introduced in the wireline networks. Congestion in wireline broadband networks can cause fluctuations in the availability of network resources, thereby resulting in severe degradation of QoS. To overcome this problem, many techniques are proposed such as the adaptation of compression parameters [73] and layered coding [81]. The much more severe resource and bandwidth fluctuations in cellular mobile networks make it interesting to consider the use of adaptive multimedia in future cellular mobile systems.

In adaptive multimedia applications, a multimedia call can dynamically change its bandwidth to adapt to the fluctuating communication environment throughout its lifetime. Assume that there are  $K$  classes of services in the network. A class  $i$  call uses bandwidth among  $\{b_{i1}, b_{i2}, \dots, b_{ij}, \dots, b_{iN_i}\}$  where  $b_{ij} < b_{i(j+1)}$  for  $i = 1, 2, \dots, K, j = 1, 2, \dots, N_i$  and  $N_i$  is the highest bandwidth level which can be used by class  $i$  calls. For example, using the layered coding technique, a raw video sequence is compressed into several layers [81], say, three layers: a base layer and two enhancement layers. The base layer can be independently decoded and it provides basic video quality; the enhancement layers can only be decoded together with the base layer and they further refine the quality of the base layer. Therefore, a video stream compressed into three layers can adapt to three levels of bandwidth usage (e.g., 64 kbps, 256 kbps, and 1Mbps).

## 4.2.2 Adaptive Cellular Mobile Networks

Compared to wireline networks, the fluctuations in resource availability in cellular mobile networks are much more severe. This stems from two inherent characteristics of cellular mobile networks: fading and mobility. The fading in a wireless channel is highly varying with time and spatial dependencies which results in a transmission link with highly varying bandwidth. Moreover, mobile users are free to move from one cell to another one. The availability of resources of the original cell does not necessarily guarantee that the resources are available in new cells. The change in network resources can result in a major fluctuation in the availability of network resources served for a call.

Due to the severe fluctuation of resources in wireless mobile networks, the ability of adapting to the communication environment is very important in future cellular mobile systems. For example, in universal mobile telecommunications system (UMTS), a radio bearer established for a call can be dynamically reconfigured during the call session [1].

Figure 4.1 shows the signaling procedure between user terminal (UE) and universal terrestrial

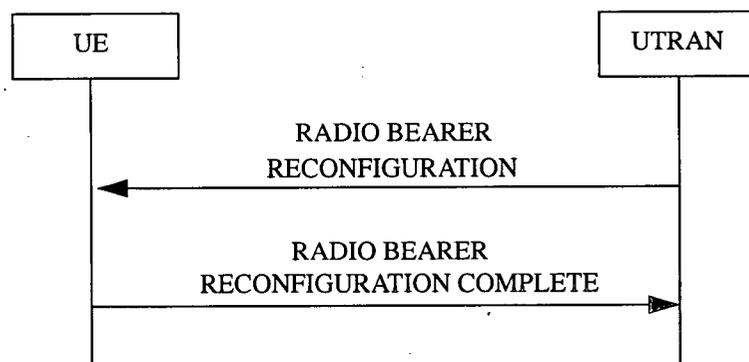


Figure 4.1 Radio bearer reconfiguration in UMTS

radio access network (UTRAN) in radio bearer reconfiguration. Radio bearer information in UMTS includes most of the information in layer 2 and layer 1 for that call, e.g., radio link control (RLC), power control, spreading factor, diversity, etc. By reconfiguring the radio bearer, the bandwidth of a call can be changed dynamically during a call session.

### 4.2.3 QoS Provisioning in Adaptive Framework

The goal of QoS provisioning in an adaptive multimedia framework is to maximize the long-term network revenue and guarantee QoS constraints. We consider two important modules for QoS provisioning, CAC and BA, in this study. When a cell is in under-loaded condition, CAC tries to accept every call and BA tries to allocate as much bandwidth as possible to each call. However, network congestion may occur and QoS constraints may be violated. In this case, calls should be rejected by CAC or degraded to a lower bandwidth by BA. On the other hand, if a call releases the bandwidth due to either call completion or handoff to another cell, some of the calls left in that cell might increase their bandwidths. To decide which call to accept and which call(s) to change the bandwidth are the roles of CAC and BA, respectively, in the adaptive multimedia framework.

To reduce network signaling overhead, we assume that the BA can only be used when a call arrival or departure occurs. That is, BA will not be used when a brief congestion occurs due to channel fading, because the reallocation of resources will be outdated when it is received by the mobile and the action taken in response to the brief congestion is likely to be ineffective. Low level mechanisms such as error correction coding and efficient packet scheduling are usually used to handle brief throughput variations of wireless links.

Smaller cells (micro/pico-cells) will be employed in future cellular mobile networks to increase capacity. Therefore, the number of handoffs during a call's lifetime is likely to be increased and the status of neighboring cells has an increased influence on the QoS of the local cell. In order to adapt to changes in traffic pattern, the status information of neighboring cells should be considered in QoS provisioning.

We consider three QoS constraints in this study. Since forced call terminations due to handoff dropping are generally more objectionable than new call blocking, an important call-level QoS constraint in wireless cellular networks is  $P_{hd}$ , the probability of handoff dropping, which is same as that in non-adaptive case considered in Chapters 2 and 3. In addition, although adaptive applications can tolerate decreased bandwidth, it is desirable for some applications to have a bound on the average allocated bandwidth. So, we need another QoS parameter to quantify the average bandwidth received by a call. The normalized average allocated bandwidth of class  $i$  calls, denoted as  $AB^i$ , is the ratio of the average bandwidth received by class  $i$  calls to the bandwidth with un-degraded service. In order to guarantee the QoS of adaptive multimedia,  $AB^i$  should also be kept above a target value. Finally, due to bandwidth adaptation, some calls may operate at very high bandwidth levels, whereas some calls within the same class may operate at very low bandwidth levels. This is undesirable from users' perspective. Therefore, QoS provisioning scheme should be fair to all calls within one class and intra-class fairness is another QoS constraint. These constraints will be formulated in Section 4.4.

We formulate the QoS provisioning problem as a semi-Markov decision process

(SMDP) [61]. There are several well-known algorithms, such as policy iteration, value iteration and linear programming [61] that find the optimal solution of SMDP. However, these traditional model-based solutions (require state transition probabilities) to SMDP suffer from two “curses”: The curse of dimensionality and the curse of modeling. These curses motivate us to pursue alternative approach to solve this problem.

#### 4.2.4 Average Reward RL

In recent year, an alternative approach called reinforcement learning has become a topic of intensive research. RL combines concepts from dynamic programming, stochastic approximation via simulation, and function approximation that may be performed with the help of regression or neural networks. This method has two distinct advantages over model-based methods. The first is that it can handle problems with complex transitions by making judicious use of stochastic approximation thereby eliminating the need to compute or store the transition probabilities. Secondly, RL can integrate within it various function approximation methods (e.g., neural networks), which can be used to approximate the value function even when the size of the state space is gargantuan.

Most of the published research in RL is focused on the discounted sum of rewards as the optimality metric. In some domains, such as economics, discounting can be used to represent “interest” earned on rewards, so that an action that generates an immediate reward will be preferred over one that generates the same reward some steps into the future. *Q*-learning [77] is one of the most popular discounted reward RL algorithms. These techniques, however, cannot extend automatically to the average reward problem, which is harder to

analyze even when the model is readily available. Not surprisingly, the corresponding developments for average reward problems have been slower. In many engineering design and decision making problems, performance measures may not suitably be described in economic terms, and hence it may be preferable to compare policies based on time averaged expected reward rather than expected total discounted reward. Some examples of such measures are average waiting time of a job in the queue, average number of customers in the system, average throughput of a machine, and average percentage of satisfied demands in an inventory system. The QoS provisioning problem in this study falls in this category. Discounted RL methods, e.g.,  $Q$ -learning, can lead to sub-optimal behavior and may converge much more slowly than average reward RL methods [52]. An algorithm for average reward RL called SMART (Semi-Markov Average Reward Technique) [26], [37], [38] has emerged recently. The convergence analysis of this algorithm is given in [37] and it has been successfully applied to production inventory [26] and airline seat allocation [38] problems.

We use this average reward RL method to solve the QoS provisioning problem for adaptive wireless multimedia in this chapter. The formulation and the performance of this method in solving QoS provisioning are presented in the following sections.

### **4.3 Average Reward RL for Solving SMDP**

#### **4.3.1 Average Reward SMDP**

For an SMDP, let  $S$  be a finite set of states and  $A$  be a set of possible actions. For  $s \in S$ , when an action  $a \in A$  is chosen, a lump sum reward of  $k(s, a)$  is received. Further accrual of reward occurs at a rate  $c(s', s, a)$ ,  $s' \in S$ , for the time the natural process remains

in state  $s'$  between the decision epochs. Note that the natural process may change state several times between decision epochs, and therefore, the rate at which the rewards are accumulated between decision epochs may vary. The expected reward between two decision epochs, given that the system is in state  $s$ , and  $a$  is chosen at the first decision epoch, may be expressed as:

$$r(s, a) = k(s, a) + E \left\{ \int_0^\tau c(W_t, s, a) dt \right\}, \quad (4.1)$$

where  $\tau$  is the transition time to the second decision epoch, and  $W_t$  denotes the state of the natural process. Let  $n_t$  denote the number of decisions made up to time  $t$ . Starting from state  $s$  at time 0 and using a policy  $\pi$ , the expected total reward generated by the process up to time  $t$  can be written as:

$$v_t^\pi(s) = E_s^\pi \left\{ \int_0^t c(w_u, s_{n_u}, a_{n_u}) du + \sum_{i=0}^{n_t-1} k(s_i + a_i) \right\}. \quad (4.2)$$

The average reward  $g^\pi$  for a policy can be given as:

$$g^\pi(s) = \lim_{N \rightarrow \infty} \frac{E_s^\pi \left\{ \sum_{n=0}^N [k(s_n, a_n) + \int_{\sigma_n}^{\sigma_{n+1}} c(W_t, s_n, a_n) dt] \right\}}{E_s^\pi \left[ \sum_{n=0}^N \tau_n \right]}, \quad (4.3)$$

where  $\sigma_n$  represents the time of the  $(n+1)$ th decision epoch and  $\tau_n = \sigma_{n+1} - \sigma_n$ .

The Bellman optimality equation for average reward SMDPs can be stated as follows [61].

**Theorem 4.1:** *Under considerations of average reward over an infinite time horizon*

for any unichain SMDP, there exists a scalar  $g^*$  and a value function  $R^*$  satisfying the system of equations for all  $s' \in S$ ,

$$R^*(s) = \max_{a \in A} \left( r(s, a) - g^* y(s, a) + \sum_{s' \in S} P_{ss'}(a) R^*(s') \right), \quad (4.4)$$

where  $y(s, a)$  is the expected sojourn time in state  $s$  when action  $a$  is taken in it, and  $P_{ss'}(a)$  is the probability of transition from state  $s$  to state  $s'$  under action  $a$  in one step.

For a proof of Theorem 4.1, see Chapter 11 of [61].

There are several model-based algorithms that can be derived from the above Bellman equation for solving SMDPs. However, the two “curses” suffered by the model-based algorithms make these model-based algorithms impractical in solving SMDP for QoS provisioning in adaptive multimedia cellular systems. Therefore, we use the model-free RL algorithm to solve the average reward SMDP, which is described in the following subsection.

### 4.3.2 Solving the Average Reward SMDP Using RL

RL is a way of teaching agents (decision makers) optimal policies by assigning rewards and punishments for their actions based on the temporal feedback obtained during active interactions of the agent with the system environment. In the RL model depicted in Figure 4.2, a learning agent selects an action for the system that leads the system along a unique path till another decision-making state is encountered. At this time, the system needs to consult with the learning agent for the next state. During a state transition, the agent gathers information about the new state, immediate reward and the time spent during the state-transition, based on which the agent updates its knowledge base using an algorithm and selects the

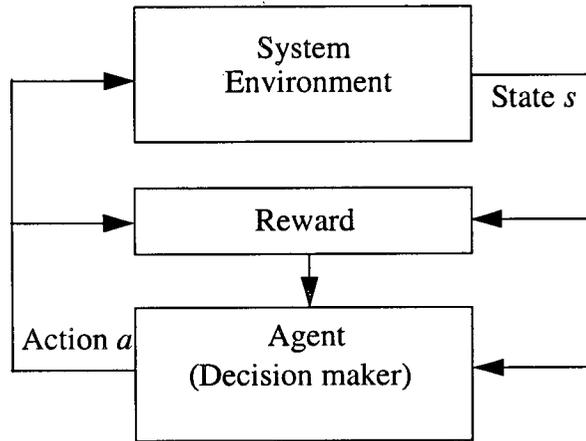


Figure 4.2 A reinforcement learning model

next action. The process is repeated and the learning agent continues to improve its performance.

Average reward RL uses the action value representation that is similar to its counterpart,  $Q$ -learning. The action value  $R^\pi(s, a)$  represents the average adjusted value of doing an action  $a$  in state  $s$  once, and then following policy  $\pi$  subsequently [52]. Let  $R^*(s, a)$  be the average adjusted value by choosing actions optimally. The Bellman equation for average reward SMDPs equation 4.4 can be rewritten as:

$$R^*(s, a) = r(s, a) - g^* + \sum_{s' \in S} P_{ss'}(a) \max_{b \in A} R^*(s', b). \quad (4.5)$$

The optimal policy is  $\pi^*(s) = \arg \max_{b \in A} R^*(s, b)$ . Since the  $R$  function makes the action explicit, the average reward RL algorithm estimates action values on-line using a temporal difference method, and then uses them to define a policy. The action value of state-action pair

$(s, a)$  visited at the  $n$ th decision making epoch is updated as follows. Assume that action  $a$  in state  $s$  results in a system transition to  $s'$  at the subsequent decision epoch, then,

$$R_{new}(s, a) = (1 - \alpha_n)R_{old}(s, a) + \alpha_n[r_{act}(s', s, a) - \rho_n\tau_n + \max_{b \in A} R_{old}(s', b)] , \quad (4.6)$$

where  $\alpha_n$  is the learning rate parameter for updating of the action value of a state-action pair of the  $n$ th decision epoch and  $r_{act}(s', s, a)$  is the actual cumulative reward earned between  $n$ th and  $(n+1)$ th decision epochs.  $\tau_n$  is the actual sojourn time between the decision epochs. In the original version of SMART [26], the reward rate,  $\rho_n$ , is shown as:

$$\rho_n = \frac{T(n-1)\rho_{n-1} + r(s', s, a)}{T(n)} , \quad (4.7)$$

where  $T(n)$  denotes the sum of the time spent in all states visited till the  $n$ th epoch. The original version of SMART does not involve a step-size (learning rate) and hence its rate of change per iteration is not controlled, which will result in the divergence behavior of this algorithm in some situations. A new version of SMART is proposed and the convergence proof is given in [37]. In the new version algorithm, the reward rate,  $\rho_n$ , is:

$$\rho_n = (1 - \beta_{n-1})\rho_{n-1} + \beta_{n-1} \frac{T(n-1)\rho_{n-1} + r(s', s, a)}{T(n)} . \quad (4.8)$$

where  $\beta_{n-1}$  is the learning rate parameter. If each action is executed in each state an infinite number of times on an infinite run and  $\alpha_n, \beta_n$  are decayed appropriately, the above learning algorithm will converge to optimality [37].

## 4.4 Formulation of QoS Provisioning in Adaptive Framework

In adaptive multimedia cellular networks, we assume that call arrivals including new and handoff call arrivals follow a Poisson distribution. Each call needs a service time that is exponentially distributed. The arrival distribution and the service distribution are independent to each other. The QoS provisioning for adaptive multimedia problem can be formulated as an SMDP. In order to utilize the average reward RL algorithm, it is necessary to identify the system state, actions, rewards, and constraints. The exploration method to guarantee the RL algorithm convergence and the method to trade off action space with state space are also described in this section.

### 4.4.1 State, Actions and Rewards

At random times an event  $e$  can occur in a cell  $c$ , where  $e$  is either a new call arrival, a handoff call arrival, a call termination, or a call handoff to a neighboring cell. At this time, cell  $c$  is in a particular configuration  $\mathbf{x}$  defined by the number of each type of ongoing calls in cell  $c$ .  $\mathbf{x} = (x_{11}, x_{12}, \dots, x_{ij}, \dots, x_{KN_k})$ , where  $x_{ij}$  stands for the number of ongoing calls of class  $i$  using bandwidth  $b_{ij}$  in cell  $c$  for  $1 \leq i \leq K$  and  $1 \leq j \leq N_i$ . Recall that  $K$  is the number of service classes in the system and  $N_i$  is the highest bandwidth level of class  $i$  defined in Section 4.2. Since the status of neighboring cells is very important for the QoS provisioning, we should consider it in the state description. The status of neighboring cells  $\mathbf{y}$  can be defined as the number of each type of ongoing calls in all neighboring cells of cell  $c$ .  $\mathbf{y} = (y_{11}, y_{12}, \dots, y_{ij}, \dots, y_{KN_k})$ , where  $y_{ij}$  stands for the number of ongoing calls of class  $i$  using bandwidth  $b_{ij}$  in all neighboring cells of cell  $c$ . The configurations and the event together determine the state,  $s =$

$(x, y, e)$ .

We assume that each cell has a fixed channel capacity  $C$  and cell  $c$  has  $M$  neighboring cells. The state space is defined by as:

$$S = \left\{ s(x, y, e) : \sum_{i=1}^K \sum_{j=1}^{N_i} x_{ij} b_{ij} \leq C; \sum_{i=1}^K \sum_{j=1}^{N_i} y_{ij} b_{ij} \leq MC \right\}.$$

The state space will increase dynamically by considering the status of neighboring cells. It will be very difficult, if not impossible, to solve the problem using model-based approaches.

When an event occurs, the agent must choose an action according to the state. An action can be denoted as:  $a = (a_a, a_d, a_u)$ , where  $a_a$  stands for the admission decision, i.e., admit ( $a_a = 1$ ) or reject ( $a_a = 0$ ),  $a_d$  stands for the action of bandwidth degradation when a call is accepted and  $a_u$  stands for the action of bandwidth upgrade when there is a departure (call termination or handoff to a neighboring cell) from cell  $c$ .  $a_d$  has the form

$$a_d = \{(d_{12}^1, \dots, d_{ij}^n, \dots, d_{KN_K}^{N_K-1}), 1 \leq i \leq K, 1 < j \leq N_i, 1 \leq n < j\},$$

where  $d_{ij}^n$  denotes the number of ongoing class  $i$  calls using bandwidth  $b_{ij}$  that are degraded to bandwidth  $b_{in}$ .  $a_u$  has the form

$$a_u = \{(u_{12}^1, \dots, u_{ij}^n, \dots, u_{KN_K-1}^{N_K}), 1 \leq i \leq K, 1 \leq j < N_i, j < n \leq N_i\},$$

where  $u_{ij}^n$  denotes the number of ongoing class  $i$  calls using bandwidth  $b_{ij}$  that are upgraded to bandwidth  $b_{in}$ .

After the action of bandwidth degradation, the configuration  $(x_{11}, x_{12}, \dots, x_{ij}, \dots, x_{KN_k})$  becomes

$$\left( x_{11} + \sum_{m=2}^{N_1} d_{1m}^1, x_{12} + \sum_{m=3}^{N_1} d_{1m}^2 - d_{12}^1, \dots, x_{ij} + \sum_{m=j+1}^{N_i} d_{im}^j - \sum_{m=1}^{j-1} d_{ij}^m, \dots, x_{KN_k} - \sum_{m=1}^{N_k-1} d_{KN_k}^m \right).$$

Similarly, after the action of bandwidth upgrade, the configuration  $(x_{11}, x_{12}, \dots, x_{ij}, \dots, x_{KN_k})$  becomes

$$\left( x_{11} - \sum_{m=2}^{N_1} u_{11}^m, x_{12} + u_{11}^2 - \sum_{m=3}^{N_1} u_{12}^m, \dots, x_{ij} + \sum_{m=1}^{j-1} u_{im}^j - \sum_{m=j+1}^{N_i} u_{ij}^m, \dots, x_{KN_k} + \sum_{m=1}^{N_k-1} u_{KN_k}^m \right).$$

Based on the action taken in a state, the network earns immediate deterministic revenue due to the carried traffic in the cell. On the other hand, in order to inform the senders and receivers of their new bandwidths in the bandwidth adaptation, extra signaling overhead is required, which will consume radio and fixed line bandwidth, as well as the battery power in the mobile. It is observed in [25], [71] that frequent bandwidth switching among different levels may consume a lot of resources and may be even worse than a large degradation ratio. Thus, there is a trade-off between the network resources utilized by the calls and the signaling and processing load incurred by bandwidth adaptation operation. We use a function to model the cost due to the action of bandwidth adaptation. The definition of the cost function depends on specific traffic, user terminal, network architecture, etc. in real networks. One intuitive definition is that the cost is proportional to the number of bandwidth adaptation operations, which is used in this study.

Let  $r_{ij}$  be the reward rate of a class  $i$  call using bandwidth  $b_{ij}$ ,  $c_a$  be the cost of one

bandwidth adaptation operation, and  $N_a(a)$  be the total number of bandwidth adaptation operations in action  $a$ . The actual cumulative reward,  $r_{act}(s', s, a)$ , between two successive decision epochs starting in state  $s$  (with action  $a$ ) and ending in state  $s'$  can be calculated as:

$$r_{act}(s', s, a) = \tau_{act}(s', s, a) \sum_{i=1}^K \sum_{j=1}^{N_i} x'_{ij} r_{ij} - N_a(a) c_a ,$$

where  $\tau_{act}(s', s, a)$  is the actual sojourn time between the decision epochs.

By formulating the cost of the bandwidth adaptation operation in the model, we can control the adaptation operation frequency effectively. Note that all ongoing calls in the cell, including those that have been degraded or upgraded, contribute to the reward  $r_{act}(s', s, a)$ . Therefore, we do not need an extra term to formulate the penalty related to the bandwidth degradation.

#### 4.4.2 Constraints

For a general SMDP with  $L$  constraints, the optimal policy for at most  $L$  of the states is randomized [6]. Since  $L$  is much smaller than the total number of states in the QoS provisioning problem considered in this study, the non-randomized stationary policy learned by RL is often a good approximation to the optimal policy [34]. To avoid the complications of randomization, we concentrate on non-randomized policies in this study.

As mentioned in Section 4.2, the first QoS constraint is related to the handoff dropping probability. Upon the  $n$ th decision epoch, the probability of a handoff call being

dropped, denoted as  $P_{hd}(s_n)$ , should be kept below a target value. Let  $TP_{hd}$  denote the target maximum allowed handoff dropping probability. The constraint associated with  $P_{hd}$  can be formulated as:

$$\lim_{N \rightarrow \infty} \frac{\sum_{n=0}^N P_{hd}(s_n) \tau_n}{\sum_{n=0}^N \tau_n} \leq TP_{hd} .$$

The Lagrange multiplier formulation relating the constrained optimization to an unconstrained optimization [11], [12] is used in this study to deal with the handoff dropping constraint. To fit into this formulation, we need to include the history information in our state descriptor. The new state descriptor is  $\bar{s} = (N_{hr}, N_{hd}, \tau, s)$ , where  $N_{hr}$  and  $N_{hd}$  are the total number of handoff call requests and handoff call drops, respectively, from each class,  $\tau$  is the time interval between the last and the current decision epochs, and  $s$  is the original state descriptor. In order to make the state space finite, quantified values of  $P_{hd} = N_{hd}/N_{hr}$  and  $\tau$  are used.

A Lagrange multiplier  $\omega$  is used for the parameterized reward

$$\bar{r}_{act}(\bar{s}', \bar{s}, a) = r_{act}(\bar{s}', \bar{s}, a) - \omega z(\bar{s}', \bar{s}, a),$$

where  $r_{act}(\bar{s}', \bar{s}, a)$  is the original reward function and  $z(\bar{s}', \bar{s}, a) = P_{hd}(\bar{s}) \tau_{act}(\bar{s}', \bar{s}, a)$  is the cost function associated with the constraint. A nice monotonicity property associated

with  $\omega$  shown in [11] facilitates the search for a suitable  $\omega$ .

The second QoS constraint is related to  $AB^i$ , the normalized average allocated bandwidth of class  $i$  calls, let  $B^i$  denote the bandwidth allocated to class  $i$  calls,  $AB^i$  can be defined as the mean of  $B^i/b_{iN_i}$  over all class  $i$  calls in the current cell. Recall that  $b_{iN_i}$  is the bandwidth of a class  $i$  call with un-degraded service.

$$AB^i = E\left\{\frac{B^i}{b_{iN_i}}\right\} = \frac{E\{B^i\}}{b_{iN_i}} = \frac{\sum_{j=1}^{N_i} x_{ij} b_{ij}}{b_{iN_i} \sum_{j=1}^{N_i} x_{ij}}, i = 1, \dots, K.$$

$AB^i$  should be kept larger than the target value  $TAB^i$ :

$$AB^i \geq TAB^i, i = 1, \dots, K.$$

The third QoS constraint is the intra-class fairness constraint, which can be characterized as the variance of  $B^i/b_{iN_i}$  over all class  $i$  calls in the current cell:

$$VB^i = \text{var}\left\{\frac{B^i}{b_{iN_i}}\right\} = \frac{\text{var}\{B^i\}}{b_{iN_i}^2} = \frac{E\{(B^i)^2\} - E^2\{B^i\}}{b_{iN_i}^2} = \frac{\sum_{j=1}^{N_i} x_{ij} \sum_{j=1}^{N_i} x_{ij} (b_{ij})^2 - \left(\sum_{j=1}^{N_i} x_{ij} b_{ij}\right)^2}{b_{iN_i}^2 \left(\sum_{j=1}^{N_i} x_{ij}\right)},$$

$$i = 1, \dots, K.$$

$VB^i$  reflects how different are the bandwidths between class  $i$  calls. For absolute

fairness,  $VB^i$  should be kept to zero all the time. However, this will be very difficult in the real situation. Therefore, it is better to keep  $VB^i$  below a target value  $TVB^i$  as well:

$$VB^i \geq TVB^i, i = 1, \dots, K .$$

$AB^i$  and  $VB^i$  are intrinsic properties of a state. With the current state and action information  $(\bar{s}, a)$ , we can forecast  $AB^i$  and  $VB^i$  in the next state  $\bar{s}'$ ,  $AB^i(\bar{s}')$  and  $VB^i(\bar{s}')$ . If  $AB^i(\bar{s}') \leq TAB^i$  and  $VB^i(\bar{s}') \leq TVB^i$ ,  $i = 1, \dots, K$ , the action is feasible. Otherwise, this action should be eliminated from the feasible action set  $A(\bar{s})$ .

### 4.4.3 Exploration

As mentioned in Section 4.3, each action should be executed in each state an infinite number of times to guarantee the convergence of RL algorithms. This is called *exploration* [10]. Exploration plays an important role in ensuring that all the states of the underlying Markov chain are visited by the system and all the potentially beneficial actions in each state are tried out. Therefore, with a small probability  $p_n$  upon the  $n$ th decision-making epoch, decisions other than that with the highest action value should be taken.

In this work, we use the Darken-Chang-Moody search-then-converge procedure [27] to decay the learning rates  $\alpha_n$ , and the exploration rate  $p_n$ . In the following expression,  $\Theta_n$  can be substituted by  $\alpha_n$ ,  $\beta_n$  and  $p_n$  for learning and exploration respectively.  $\Theta_n = \frac{\Theta_0}{1 + u}$ ,

where  $u = \frac{n^2}{\Theta_r + n}$ , where  $\Theta_0$  and  $\Theta_r$  are constants.

#### 4.4.4 Trading off Action Space Complexity with State Space Complexity

We can see that the action space in our formulation is quite large. It will be very time-consuming to find the suitable action given a specific state using RL. As mentioned in [82], the large action space problem may hinder the deployment of the scheme proposed in [82] in real networks where the number of bandwidth levels is usually large. In this work, we propose a method to trade off action space complexity with state space complexity in the QoS provisioning scheme using an algorithm described in [10]. The advantages of doing this are that the action space will be reduced and the extra state space complexity may still be dealt with by using the function approximation described in Section 4.5.

Suppose that a call arrival event occurs in a cell with state  $s$ , the action that can be chosen from is  $a = (a_a, d_{12}^1, \dots, d_{ij}^n, \dots, d_{KN_k}^{N_k-1})$ , where there are at most

$V = 1 + \sum_{i=1}^K \sum_{j=2}^{N_i} (j-1)$  components. We can break down the action  $a$  into a sequence of  $V$

controls  $a_a, d_{12}^1, \dots, d_{ij}^n, \dots, d_{KN_k}^{N_k-1}$ , and introduce some artificial intermediate “states”  $(\bar{s}, a_a)$ ,

$(\bar{s}, a_a, d_{12}^1), \dots, (\bar{s}, a_a, d_{12}^1, \dots, d_{ij}^n, \dots, d_{KN_k}^{N_k-1})$ , and the corresponding transitions to model the

effect of these actions. In this way, the action space is simplified at the expense of introducing

$V-1$  additional layers of states and  $V-1$  additional action values  $R(\bar{s}, a), R(\bar{s}, a_a, d_{12}^1), \dots,$

$R(\bar{s}, a_a, d_{12}^1, \dots, d_{ij}^n, \dots, d_{KN_k}^{N_k-2})$  in addition to  $R(\bar{s}, a_a, d_{12}^1, \dots, d_{ij}^n, \dots, d_{KN_k}^{N_k-1})$ . Actually, we view

the problem as a deterministic dynamic programming problem with  $V$  stage. For  $v = 1, \dots, V$ , we can have an  $v$ -solution (a partial solution involving just  $v$  components) for the  $v$ th stage of the problem. The terminal state corresponds to the  $V$ -solution (a complete solution with  $V$  components). Moreover, instead of selecting the controls in a fixed order, it is possible to leave order subject to choice.

In the reformulated problem, at any given state  $\bar{s} = (N_{hr}, N_{hd}, \tau, \mathbf{x}, \mathbf{y}, e)$  where  $e$  is a call arrival of class  $i$ , the control choices are:

1. Reject the call, in which case the configuration  $\mathbf{x}$  does not evolve.
2. Admit the call and no bandwidth adaptation is needed, in which case the configuration  $\mathbf{x}$  evolves to  $(x_{11}, x_{12}, \dots, x_{ij}, \dots, x_{iN_i} + 1, \dots, x_{KN_k})$ .
3. Admit the call and bandwidth adaptation is needed. In this case, the problem can be divided into  $V$  stages. On the  $v$ th stage ( $v = 1, \dots, V$ ), one particular call type that has not been selected in previous stages, say the one using bandwidth  $b_{ij}$  with  $x_{ij} > 0$ , can be selected and there are following options:
  - a. Degrade one call using bandwidth  $b_{ij}$  one level, in which case the configuration  $\mathbf{x}$  evolves to  $(x_{11}, x_{12}, \dots, x_{ij-1} + 1, x_{ij} - 1, \dots, x_{iN_i} + 1, \dots, x_{KN_k})$ .
  - b. Degrade two calls using bandwidth  $b_{ij}$  one level, in which case the configuration  $\mathbf{x}$  evolves to  $(x_{11}, x_{12}, \dots, x_{ij-2} + 2, x_{ij} - 2, \dots, x_{iN_i} + 1, \dots, x_{KN_k})$ .
  - c. Increase the number of calls being degraded until the call arrival can be accommodated. Please note that the number of options depends on specific selected call type and the class of call arrival.

The similar trade-off can be done when a call departure event occurs.

## 4.5 Implementation Considerations

### 4.5.1 Approximate Representation of Action Values

In practice, an important issue is how to store the action value  $R(s, a)$  in cellular mobile networks. There are several approaches, among which the lookup table is the most straightforward method to represent action values. A lookup table representation means that a separate variable  $R(s, a)$  is kept in memory for each state-action pair  $(s, a)$ . This method will be prohibitive when the number of state-action pairs becomes large because memory requirement can be huge. Approximate representation should be used to break the curse of dimensionality in the face of very large state spaces. A neural network is an efficient method to represent the action values. A common popular neural network architecture is the *multi-layer perceptron (MLP) with a single hidden layer* [10] as shown in Figure 4.3. Under this architecture, the state-action pair  $(s, a)$  is encoded as a vector and transformed linearly through the input layer involving coefficients in this layer to give several scalars. Then, each of these scalars becomes the input to the sigmoidal function in the hidden layer. Finally, the outputs of the sigmoidal functions are linearly combined using coefficients to produce the final output. These coefficients are also known as *weights* of the network.

The network is trained in a supervised fashion using the back-propagation algorithm. This means that during training both network inputs and target outputs are used. An input pattern is applied to the network and an output is generated. This output is compared to the corresponding target output and an error is produced and propagated back through the

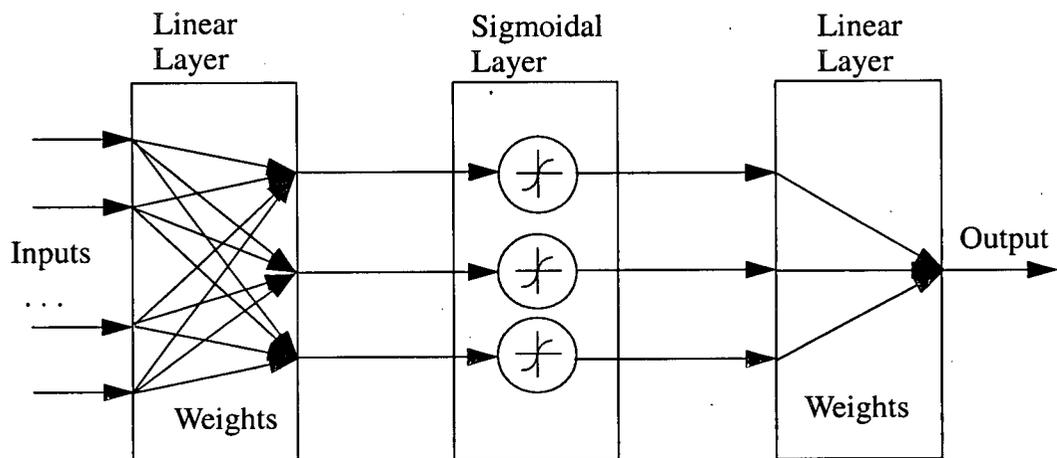


Figure 4.3 The neural network used in approximation

network, and the network weights are adjusted to minimize the sum of the errors squared. The procedure is repeated until the error is reduced to a sufficient level.

#### 4.5.2 Structure and Pseudo-code

The structure of the RL-based QoS provisioning scheme is shown in Figure 4.4. First of all, an event (either a call arrival or a call departure) occurs and a state  $s$  can be identified by getting the status of the local cell and neighboring cells. Then, a set of actions  $\{a\}$  can be found according to the state. Feed the state and action information into the neural network to get the action values. With probability  $1 - p_n$ , choose the action with the largest action value. Otherwise, perform the exploration, i.e., choose an action randomly. When the next event occurs, the action value is updated and the process is repeated. A pseudo-code description of the proposed scheme is given in Figure 4.5.

#### 4.6 Simulation Results and Discussions

A cellular network of 19 cells is used in our simulations, as shown in Figure 4.6. To

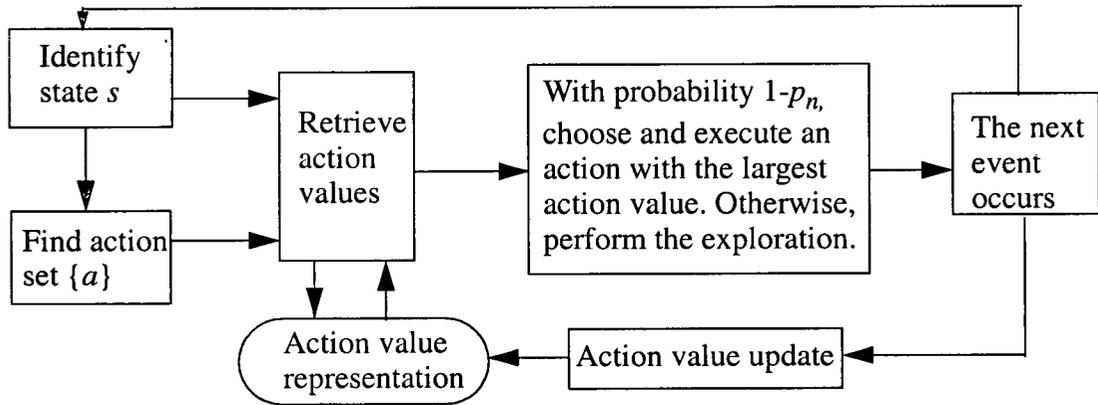


Figure 4.4 The structure of the QoS provisioning scheme

```

initialize iteration count  $n := 0$ , action value  $R(s, a) := 0$ ,
cumulative reward  $CR := 0$ , total time  $T := 0$ ,
reward rate  $\rho_0 := 0$ 
while  $n < \text{MAX\_STEPS}$  do
  calculate  $p_n, \alpha_n, \beta_n$  using iteration count  $n$ 
  with probability of  $(1 - p_n)$ , tradeoff action space with state
  space and choose an action  $a_n \in A$  that maximizes  $R(s_n, a_n)$ .
  Otherwise, choose a random (exploratory) action from  $A$ 
  execute the chosen action
  wait for the next event  $e$ 
  update  $R_{new}(s_n, a_n) = (1 - \alpha_n)R_{old}(s_n, a_n) +$ 

$$\alpha_n \left[ r(s_{n+1}, s_n, a_n) - \rho_n \tau(s_{n+1}, s_n, a_n) + \max_{b \in A} R_{old}(s_{n+1}, b) \right]$$

  if the action value is stored in neural network
     $R_{new}(s_n, a_n) - R_{old}(s_n, a_n)$  is served as an back
    propagated error to learn the weight parameters in
    the neural network
  endif
  if an exploration action was not chosen
    update  $CR = CR + r(s_{n+1}, s_n, a_n)$ ,
 $T_{n+1} = T_n + \tau(s_{n+1}, s_n, a_n)$  and

$$\rho_{n+1} = (1 - \beta_n)\rho_n + \beta_n \frac{T_n \rho_n + r(s_{n+1}, s_n, a_n)}{T_n}$$

  endif
  update iteration count  $n = n + 1$ 

```

Figure 4.5 A pseudo-code of the QoS provisioning scheme

avoid the edge effect of the finite network size, wrap-around is applied to the edge cells so that each cell has six neighbors. For example, cells 2, 9, 10, 14, 18 and 19 are the neighbors of

cell 8. Each cell has a fixed bandwidth of 2 Mbps. Two classes of flows are considered (see Table 4.1). Class 1 traffic has three different bandwidth levels, 128, 192 and 256 kbps. 64, 96

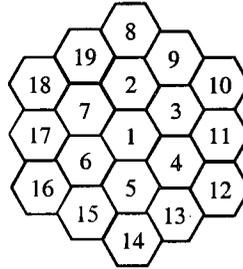


Figure 4.6 A cellular network used in simulations

and 128 kbps are the three possible bandwidth levels of class 2 traffic. Two reward functions are used in simulations, as shown in Table 4.1. Reward function 1 represents the scenario that the reward generated by a call is a linear growing function with the bandwidth assigned to the call. Specifically,  $r_{ij} = b_{ij}$ . In reward function 2, a convex function  $r_{ij} = \frac{b_{max}^2 - (b_{ij} - b_{max})^2}{b_{max}}$

is used, where  $b_{max}$  is largest bandwidth used by a call in the network. We assume that the highest possible bandwidth level is requested by the call arrival. That is, call arrival of class 1 always requests 256 kbps and call arrival of class 2 always requests 128 kbps. Then the

Table 4.1 Experimental parameters

Traffic Class	Bandwidth Level (kbps)	Reward Function 1	Reward Function 2
Class 1	$b_{11}$ : 128	$r_{11}$ : 128	$r_{11}$ : 192
	$b_{12}$ : 192	$r_{12}$ : 192	$r_{12}$ : 240
	$b_{13}$ : 256	$r_{13}$ : 256	$r_{13}$ : 256

Table 4.1 Experimental parameters

Traffic Class	Bandwidth Level (kbps)	Reward Function 1	Reward Function 2
Class 2	$b_{21}$ : 64	$r_{21}$ : 64	$r_{21}$ : 112
	$b_{22}$ : 96	$r_{22}$ : 96	$r_{22}$ : 156
	$b_{23}$ : 128	$r_{23}$ : 128	$r_{23}$ : 192

network will make the CAC decision and decide which bandwidth level the call can use if it is admitted. 30% of the offered traffic is from class 1. Moreover, call holding time and cell residence time are assumed to follow exponential distributions with mean values 180 seconds and 150 seconds, respectively. The probability of a user handing off to any adjacent cell is equally likely. The target maximum allowed handoff dropping probability,  $TP_{hd}$ , is 1% for both classes. Other QoS parameters are changed in the simulations.

The action values are learnt by running the simulation for 30 million steps with a constant new call arrival rate of 0.1 calls/cell/second. The constants used for the learning and exploration rates are chosen as  $\alpha_0 = \beta_0 = p_0 = 0.1$ , and  $\alpha_r = \beta_r = p_r = 10^{-11}$ , which are common values in the Darken-Chang-Moody decaying scheme [27]. The monotonicity property associated with the Lagrange multiplier shown in [11] is used to search a suitable  $\omega$ , which is 157560 in the simulations. A multi-layer neural network is used in the approximate representation of action values. There are 31 input units representing the state and action, 20 hidden units with sigmoid functions, and one output unit representing the action value. Among the 31 input units, one unit represents the event; six units represent the status of the local cell and six units represent the status of neighboring cells; one unit represents the

handoff dropping probability; one unit represents the time interval between the last and the current decision epochs; 13 units represent the action; one unit represents the number of bandwidth adaptation operations; one unit represent the Lagrange multiplier; one unit is the bias unit that always has an input of one. The neural network is trained on-line by using the back-propagation algorithm in conjunction with the reinforcement learning.

Two QoS provisioning schemes are used for comparisons, guard channel (GC) scheme [41], [60] for non-adaptive traffic and ZCD02 scheme [92] for adaptive multimedia. 256 kbps is reserved for handoff calls in the GC scheme. In ZCD02, an optimal call mix selection scheme is derived using simulated annealing. The proposed scheme is called RL in the following. The linear reward function is used in all simulation experiments unless otherwise mentioned.

We first use uniform traffic in simulations, where the traffic load is the same among all 19 cells. Call arrivals of both classes to each cell form a Poisson process with mean  $\lambda$ .

The average rewards of different schemes normalized by the GC scheme are shown in Figure 4.7. Average allocated bandwidth and intra-class fairness constraints are not considered here. We can see that RL and ZCD02 yield more rewards than GC. In GC scheme, bandwidth adaptation is not used and a call will be rejected if no free bandwidth is available. Both RL and ZCD02 have bandwidth adaptation function, and therefore, can yield more rewards than GC. In Figure 4.7, the reward of the proposed scheme is similar to that in ZCD02, because both of them can maximize network revenue in QoS provisioning. We can also observe that at low traffic load, as the new call arrival rate increases, the gain becomes

more significant. This is because the heavier the offered load, the more the bandwidth adaptation is needed when the cell is not saturated. However, when the traffic is high and the cell is becoming saturated, the performance gain of RL and ZCD02 over GC is less significant. The cost of adaptation operation is not considered, i.e.,  $c_a = 0$ , in Figure 4.7.

Figure 4.8 shows the effects of  $c_a$ , the cost of adaptation operation, when new call arrival rate is 0.067 calls/cell/second. The reward of ZCD02 drops quickly as  $c_a$  increases, and even becomes less than that in GC when  $c_a = 150$ . In contrast, the reward drops slowly in RL. Since the proposed scheme formulates  $c_a$  in the reward function, it eliminates those actions requiring a large number of adaptation operations when  $c_a$  is high by comparing the action values of different actions. Therefore, the proposed scheme can control the adaptation cost, and therefore, the adaptation frequency, effectively. We use  $c_a = 30$  in the following simulation experiments.

Figure 4.9 shows that RL maintains an almost constant handoff dropping probability for a large range of new call arrival rates. In contrast, neither ZCD02 nor GC can enforce the QoS guarantee for the handoff dropping probability. We can reduce the handoff dropping probability in GC scheme by increasing the number of guard channels and in ZCD02 by increasing the “virtual gain function” of handoff calls. However, this will further reduce the reward earned in these two schemes. Figures 4.10 and 4.11 show the new call blocking probabilities of class 1 and class 2 traffic, respectively. Both ZCD02 and RL have less blocking probabilities compared with GC, because both of them have adaptation capability and can accept more new calls.

Figures 4.12 and 4.13 show the normalized allocated average bandwidths of class 1 and class 2 traffic, respectively.  $TAB^i = 0.7$  is considered here. We can observe that as the new call arrival rate increases, the average bandwidths of both classes in ZCD02 and RL decrease. This is the result of the bandwidth adaptation. For some applications, it may be desirable to have a bounded average allocated bandwidth. In Figures 4.12 and 4.13, it is shown that the normalized allocated average bandwidth can be bounded by the target value in RL. In contrast, ZCD02 cannot guarantee this average bandwidth QoS constraint. The average bandwidth of GC is always 1, because no adaptation operation is done in GC. Please note that the lowest possible normalized average bandwidth is 0.5 for both classes. This can be seen from Table 4.1, where the lowest bandwidth level is half of the highest bandwidth level for both classes. The normalized bandwidth variance,  $TVB$ , an indicator of intra-class fairness, is shown in Figures 4.14 and 4.15. We can see that RL can keep the bandwidth variance below the target value. Since the bandwidth in GC cannot be changed, the bandwidth variance is always 0 in GC. It can be seen that ZCD02 has very large bandwidth variance, which means that intra-class fairness cannot be satisfied in ZCD02. The achievements of higher QoS requirements come at a cost to the system. The effects of different values of  $TAB$  and  $TVB$  on the average reward are shown in Figures 4.16 and 4.17, respectively. We can see that higher  $TAB$ , which is preferred from users' point of view, will reduce the reward. Similarly, lower  $TVB$ , which means higher intra-class fairness, will reduce the reward as well.

We also consider a non-uniform traffic situation, where the cells in the second ring,

i.e., cell 2-7 in Figure 4.6, have 1.5 times the new call arrival rate in the outer ring, i.e., cell 8-19. The central cell has 2 times the new call arrival rate in the outer ring. Since the method of predicting handoff rate from neighboring cells is not given in ZCD02, a static predicted handoff rate is used in the revenue function, and we call it ZCD02-static. Figure 4.18 shows that RL yields more rewards than ZCD02-static and GC schemes. The performance gain of RL over GC and the difference between RL and ZCD02-static are significant in the non-uniform traffic situation. This is because our RL method takes into account the status of neighboring cells, and therefore, it can dynamically adapt to different traffic patterns.

The traffic load in cellular networks is typically time varying. A typical traffic pattern [32] during a 24 hour business day is shown in Figure 4.19. The peak hours occur around 11:00 a.m. and 4:00 p.m. Figure 4.20 plots the normalized average rewards under the assumption that the traffic loads are both non-uniformly distributed and temporally varying (maximum 0.067 calls/second). The reward is calculated on an hour-by-hour basis. The improvement of the proposed scheme over ZCD02-static is apparent.

We then use a convex reward function  $r_{ij} = \frac{b_{max}^2 - (b_{ij} - b_{max})^2}{b_{max}}$  in the simulations.

The reward rate for specific bandwidth level of each class is shown in Table 4.1. The simulation results using the convex reward function show a very similar pattern to those using the linear reward function, and therefore, only one figure is provided here. Figure 4.21 shows the average rewards of different QoS schemes with non-uniform traffic. We can see that Figure 4.21 is similar to Figure 4.18.

ZCD02 uses simulated annealing to find the optimal call-mix, in which a variable called *temperature* is decreased periodically by employing a monotone descendent cooling function. We follow the example given in ZCD02, where 90 temperature steps are used and each step repeats 100 times. In each of 9000 steps, the revenue and the constraints are re-evaluated. In RL, since neural network is used in the approximate representation, the major operations required to make the CAC and BA decisions come from retrieving action values and comparing these action values. We run the simulations with a fixed call arrival rate of 0.1 calls/second for 1000 call arrivals and departures, and calculate the average number of operations (additions, multiplications and comparisons) required to make one decision. Table 4.2 shows the average number of operations. We observe that ZCD02 needs many more operations than RL to make the CAC and BA decisions. This shows that ZCD02 will be more expensive than RL for computation resources in practice. However, training is needed for the RL approach, whereas ZCD02 and GC do not need any training.

## 4.7 Summary

In this chapter, we have proposed a QoS provisioning scheme for adaptive multimedia in cellular mobile networks. By considering the status of neighboring cells, the proposed scheme can dynamically adapt to the changes in traffic condition. The rapid growth in the number of states and the difficulty to estimate the state transition probabilities in practical cellular systems motivate us to choose a model free average reward reinforcement learning solution to solve this problem. A method to trade off action space with state space has been proposed to solve the large action space problem in QoS provisioning. Three QoS constraints, handoff dropping probability, average allocated bandwidth and intra-class fairness are consid-

ered in the scheme. Simulation results have been presented to show the effectiveness of the proposed scheme in adaptive multimedia cellular networks.

Table 4.2 Number of numerical operations

QoS Schemes	Number of numerical Operations
ZCD02	181,240
RL	9,872

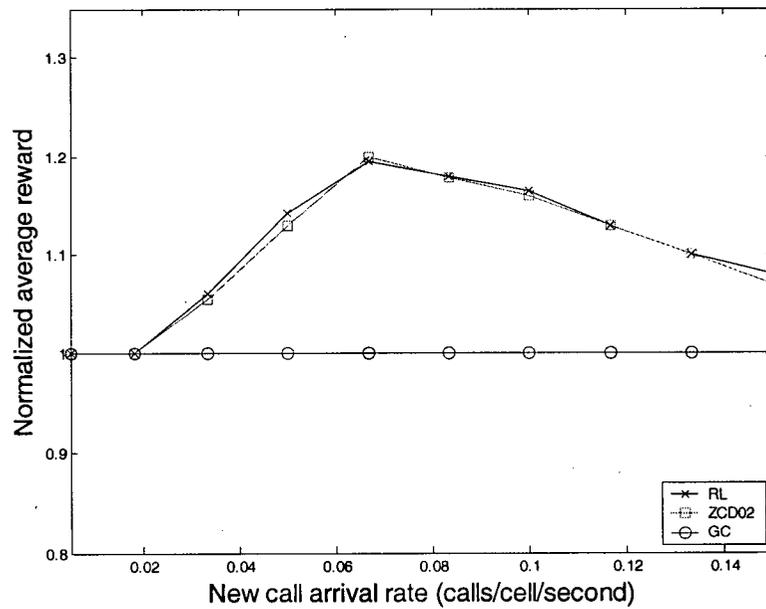


Figure 4.7 Normalized average rewards

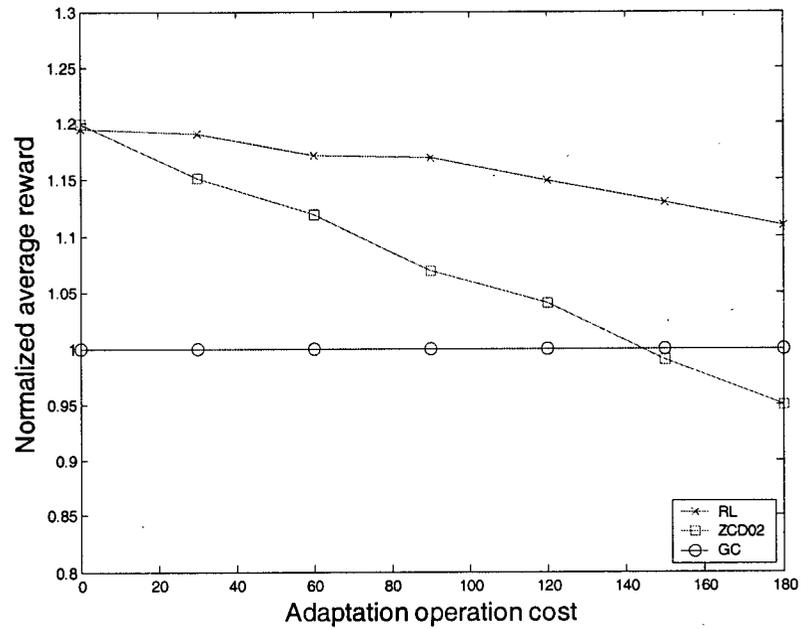


Figure 4.8 Normalized average rewards vs. adaptation operation cost

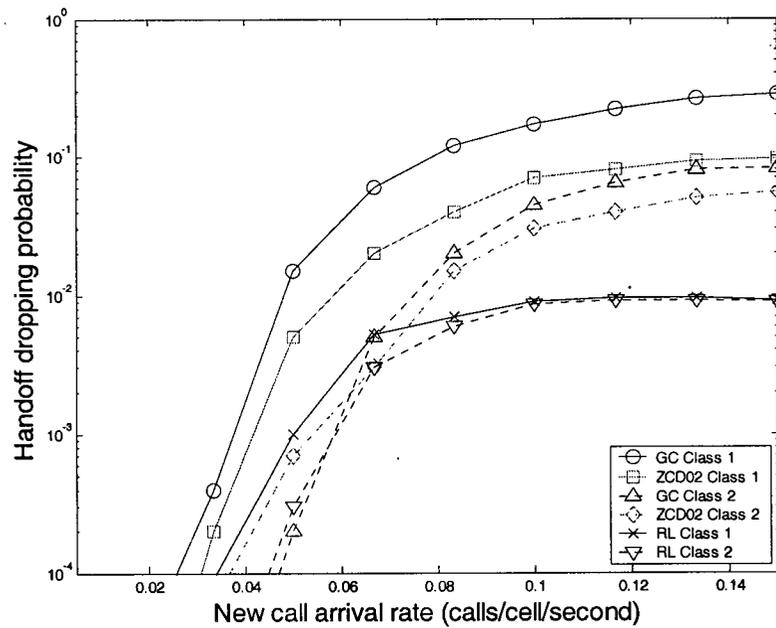


Figure 4.9 Handoff dropping probabilities

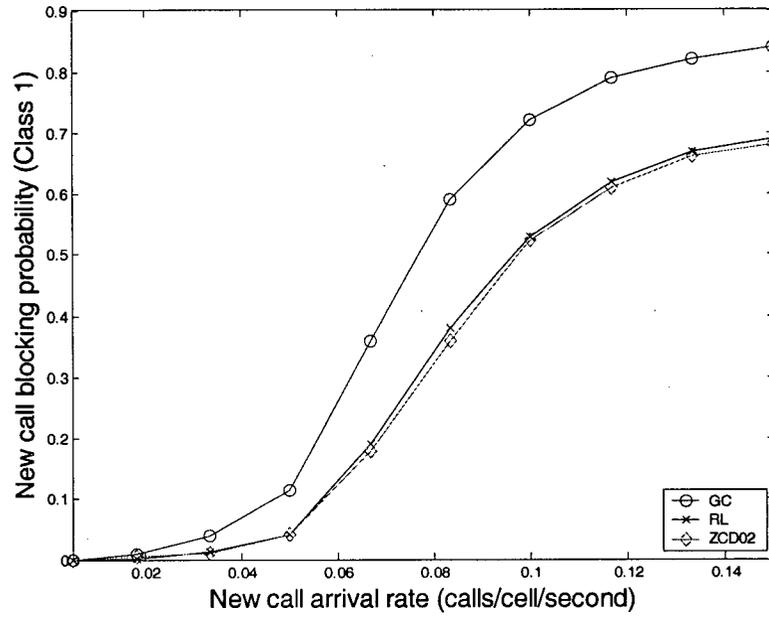


Figure 4.10 New call blocking probabilities of class 1 calls

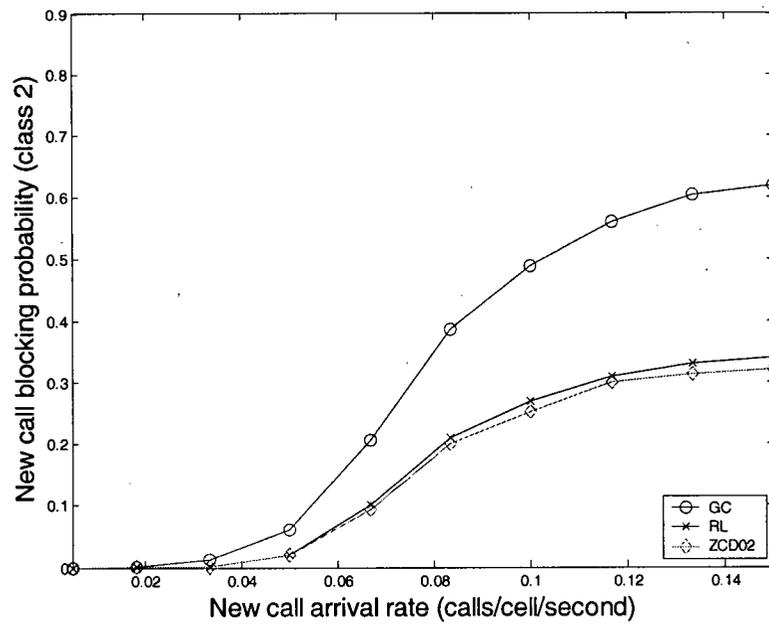


Figure 4.11 New call blocking probabilities of class 2 calls

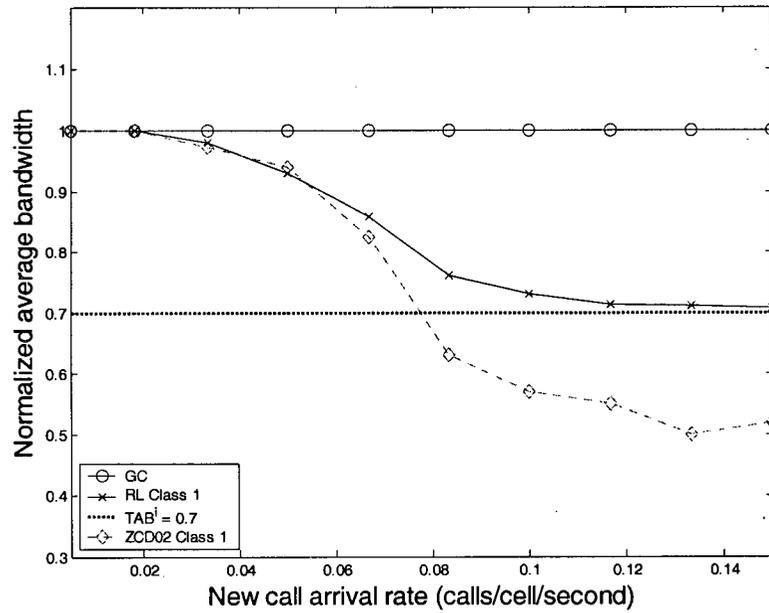


Figure 4.12 Normalized average bandwidths of class 1 calls

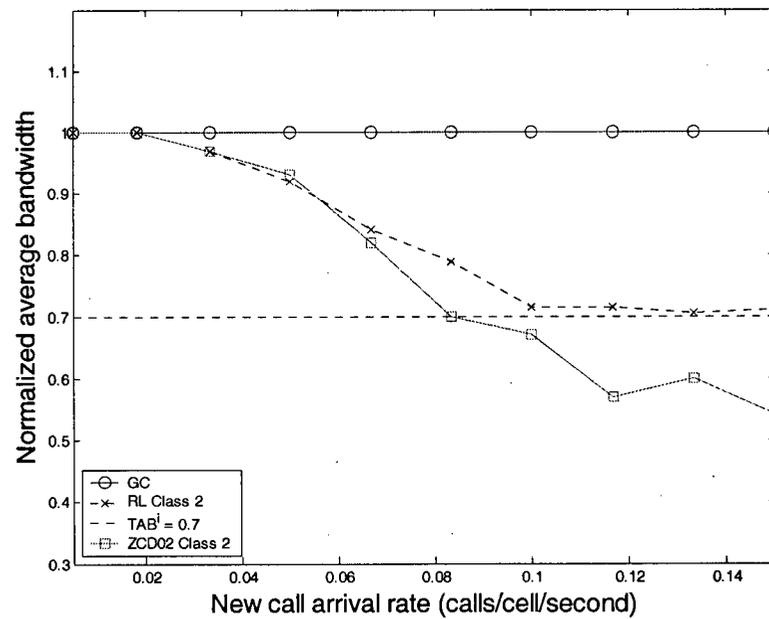


Figure 4.13 Normalized average bandwidths of class 2 calls

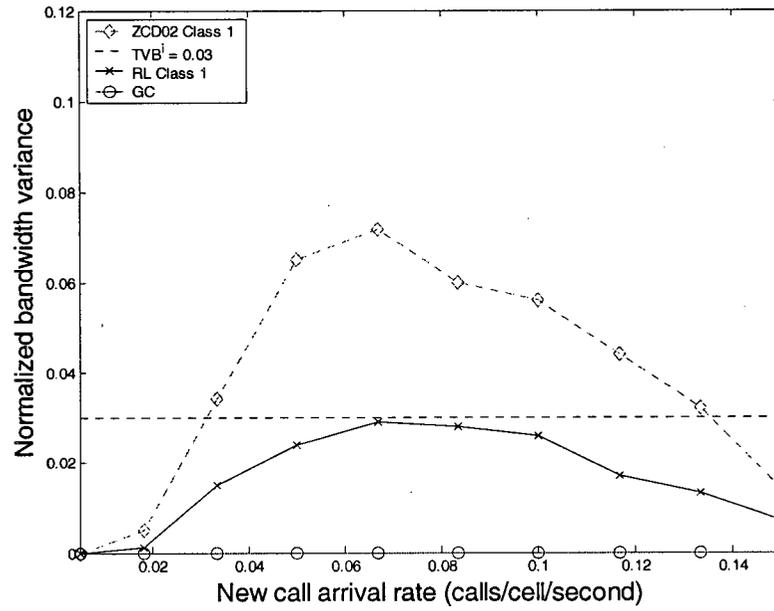


Figure 4.14 Normalized bandwidths variance of class 1 calls

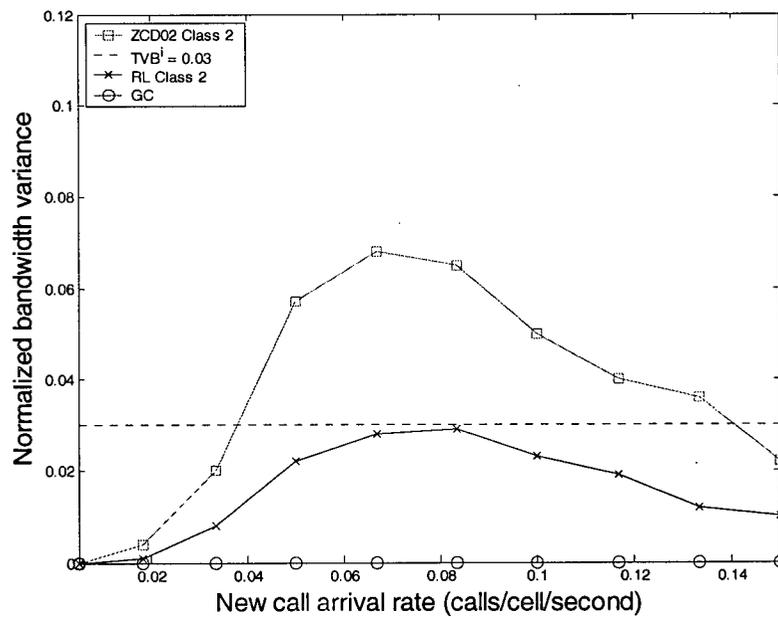


Figure 4.15 Normalized bandwidths variance of class 2 calls

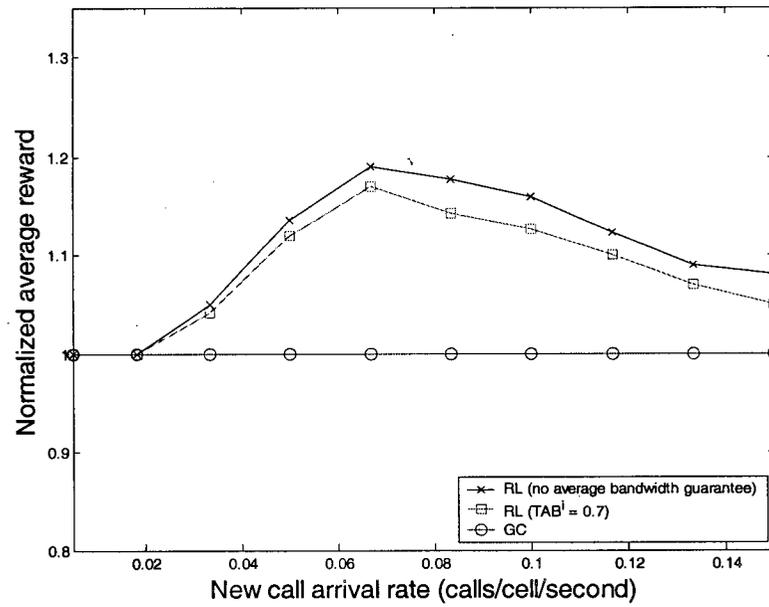


Figure 4.16 Normalized average rewards for different average bandwidth requirements

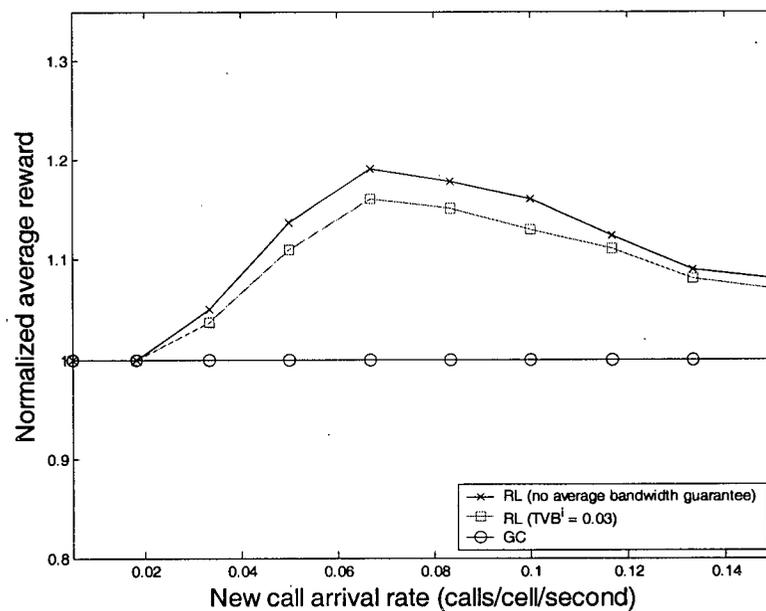


Figure 4.17 Normalized average rewards for different bandwidth variance requirements

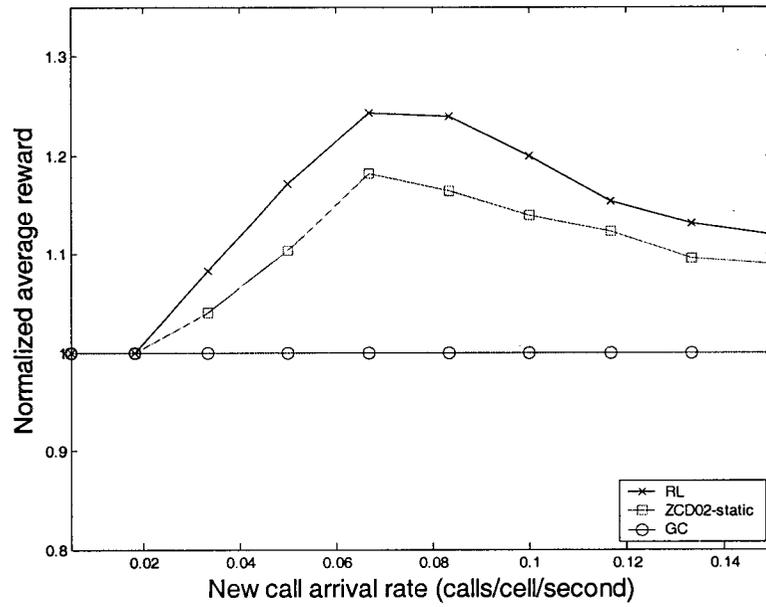


Figure 4.18 Normalized average rewards with non-uniform traffic

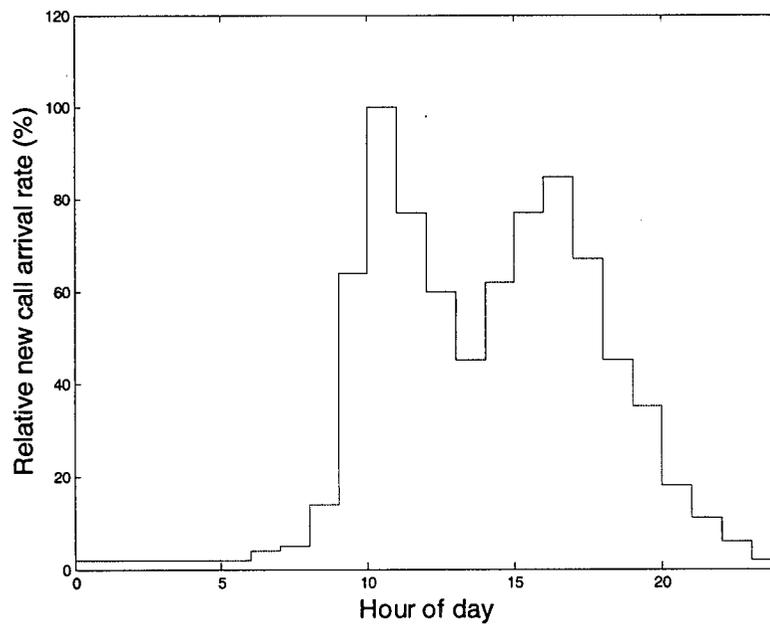


Figure 4.19 A traffic pattern of a typical business day

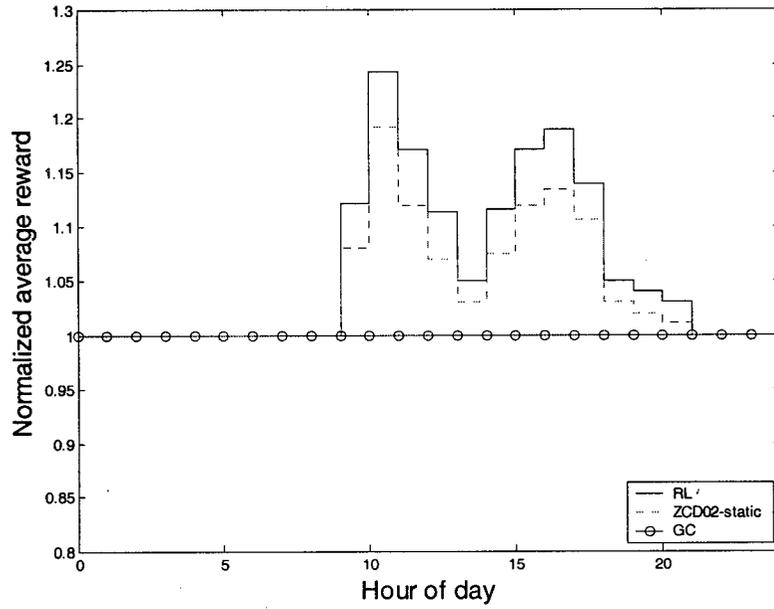


Figure 4.20 Normalized average rewards with time varying traffic

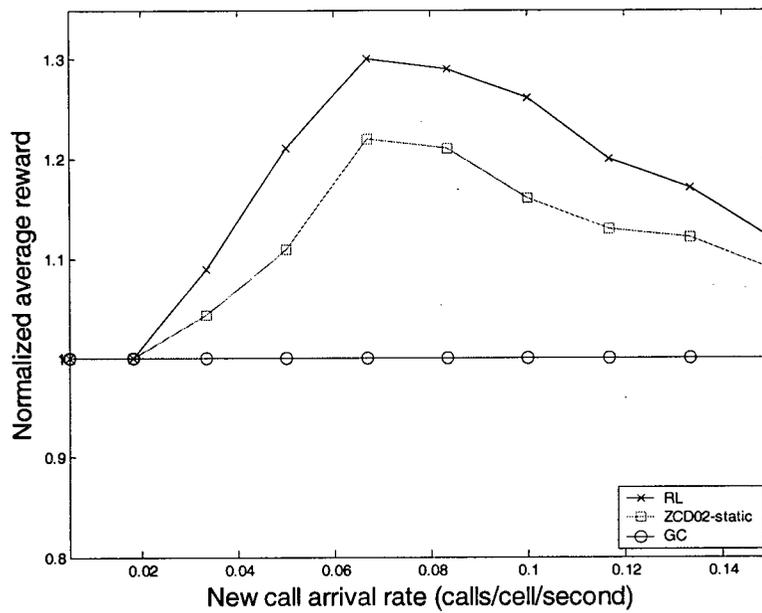


Figure 4.21 Normalized average rewards with convex reward function and non-uniform traffic

## Chapter 5 M-EMAC for Cellular-to-IP Internetworking

### 5.1 Introduction

In this chapter, we address the issue of end-to-end QoS provisioning in cellular-to-IP internetworking. When a mobile/wireless user access the Internet via a cellular network, the traffic passes through two networks: cellular network and Internet. In these two networks, they have different QoS definitions and mechanisms. However, the QoS experienced by users is end-to-end, i.e., from the server to the mobile host. Therefore, effective internetworking schemes between cellular networks and the Internet are very important to guarantee the end-to-end QoS. Particularly, we study the EMAC scheme that is proposed to enable QoS in the Internet over wireline networks. Although EMAC has many desirable features in the wireline networks as shown in previous work, in this chapter, we will show that several distinct characteristics in cellular mobile networks make EMAC difficult to implement. On the other hand, cellular mobile network, e.g., UMTS, has been designed to include its own call admission control functions. We propose a mobile EMAC (M-EMAC) scheme for cellular-to-IP internetworking [85], [86]. Simulation results show that M-EMAC outperforms EMAC in the cellular mobile domain.

The rest of this chapter is organized as follows. The EMAC scheme is presented in Section 5.2. Section 5.3 describes the performance of EMAC in cellular mobile networks. M-EMAC is proposed and its performance is evaluated in Section 5.4. Finally, we conclude this chapter in Section 5.5.

## 5.2 EMAC Scheme for the Internet

The basic feature of EMAC proposed in previous work is to use end-to-end measurements to determine whether the QoS requested by a new call can be supported over the network, and if so to accept the new call. Figure 5.1 shows the two phases of a call: a probing phase and, if the call is accepted, a data transmission phase. First the sender that wants to set up a real-time call probes the network by transmitting a stream of fixed-length packets at constant intervals, at the data rate that it wishes to reserve over the network. Upon reception of the first probing packet, the receiver starts measuring the probing packets' arrival statistics over a measurement period  $\tau$ . The measured statistics can be as simple as the average rate received during the period  $\tau$ , or more complex including delay and jitter statistics. At the end of the measurement period, the receiver estimates whether there are enough resources avail-

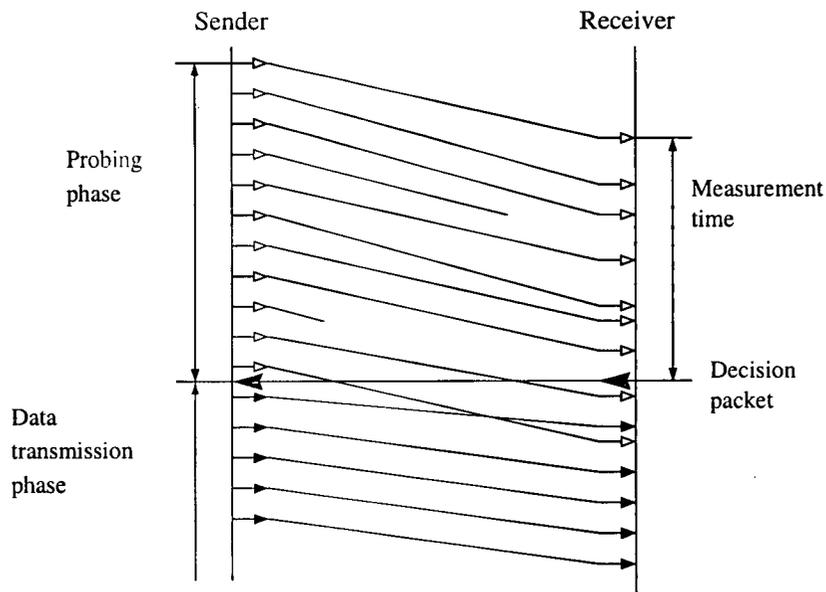


Figure 5.1 EMAC scheme for the Internet

able along the call path to meet a predetermined QoS requirement based on the statistics collected. The sender is notified of this decision to start the data transmission phase or abort the call.

In EMAC schemes, core routers are stateless and they need only to discriminate between classes of packets, in full agreement with the DiffServ paradigm. As observed in [19], there is a *bandwidth stolen* problem when fair queueing is used to service admission-controlled traffic; i.e., the bandwidth of a call having a successful probing phase may be *stolen* by subsequent arrivals. So, one should not use fair queueing or its variants to service admission-controlled traffic. Moreover, the admission-controlled traffic will coexist with current best-effort traffic and it is necessary to provide isolation between them. So, one must use scheduling mechanisms with strict rate limits on the admission-controlled flows. Based on these considerations, we use priority queueing with a rate limiter [93] to separate admission-controlled flows from best-effort ones, with FIFO service within the admission-controlled traffic itself in the following study.

Moreover, there are two choices in probing [19]: (a) send the probe packets at a lower priority than the admission-controlled data traffic (but still at a higher priority than best-effort traffic), called *out-of-band* probing; and (b) send the probe packets at the same priority as data traffic, called *in-band* probing. We will study both of these choices in our simulations.

### 5.3 EMAC in Cellular Mobile Networks

Similar to TCP, EMAC has been designed to operate over wireline networks, where the hosts are fixed, the bit error rate (BER) is very low and packet losses occur mostly

because of congestion. However, in cellular mobile networks, the scarce bandwidth, higher BER on wireless links and host mobility combine to make the implementation of EMAC challenging. We study these issues in the following subsections.

### **5.3.1 Low Bandwidth**

Compared to the bandwidth of a typical Ethernet which is around 10 Mbps (100 Mbps for fast Ethernet), the link capacity in cellular mobile networks is very scarce. For example, 3G systems such as UMTS offer only about 380 kbps for high mobility users requiring wide-area coverage. For EMAC to work well in such networks, the loss probability experienced by the probe packet in the probing phase must be a good predictor of the loss probability experienced by the data in the data transmission phase. We study this relationship in UMTS networks by simulations.

In the simulation models, mobile users communicate with servers elsewhere through a wireless access network. We assume the majority of the traffic is sent from servers outside the cellular access network to the mobile users, and the sole congestion point of the system is the forward radio link. For simplicity, the amount of uplink traffic is assumed to be negligible and have no queueing delay. The link layer assumptions in the simulation follow the specifications for Enhanced Data rate for GSM Evolution (EDGE) [33], which is one of the radio access technologies in UMTS. Without going into greater detail, one can assume that there is one time frame every 20 msec, divided into 8 time-slots. A user is allowed to occupy one or multiple time-slots in each frame. EDGE also defines eight different combinations of modulation and coding schemes, which have different data rates and robustness. A link adaptation

algorithm tries to select the most effective combination based on current channel conditions. For this study, we fix the modulation and coding scheme to the one with the second highest rate. It allows 112 bytes per time slot (excluding header), which results in a peak data rate of 44.8 kbytes/sec (358 kbps) [33]. In addition to using fixed modulation and coding scheme, we also assume that there is no packet error over the radio link in the current study, to investigate the effects of bandwidth difference between wireless and wireline networks on EMAC.

In addition, we have also developed a similar wireline model for comparisons. The link bandwidth in the wireline model is 10 Mbps. The traffic in both models are generated by *on-off* traffic sources, as shown in Tables 5.1 and 5.2. One source has exponentially distributed on and off times with a mean of 1 second and the lifetime is exponentially distributed with a mean of 300 seconds. Another one has on and off times generated by a Pareto distribution with a mean of 1 second and the lifetime is taken from a lognormal distribution with a median of 300 seconds following. Due to the lower data rate in the cellular network, we expect mobile users to be more cautious about downloading very large files. So, we set the

Table 5.1 Simulation parameters of traffic source 1

Model	Burst Rate	On Time	Off Time	Average Rate	Lifetime
Wireless	80 kbps	1 s (exp.)	1 s (exp.)	40 kbps	300 s (exp.)
Wireline	512 kbps	1 s (exp.)	1 s (exp.)	256 kbps	300 s (exp.)

Table 5.2 Simulation parameters of traffic source 2

Model	Burst Rate	On Time	Off Time	Average Rate	Lifetime
Wireless	80 kbps	1 s (Pareto)	1 s (Pareto)	40 kbps	300 s (logn.)
Wireline	512 kbps	1 s (Pareto)	1 s (Pareto)	256 kbps	300 s (logn.)

average data rate to a lower value of 40 kbps in the wireless model and 256 kbps in the wireline model. Session arrivals are generated according to a Poisson process.

In our simulations, only packet loss but not delay due to the network congestion is considered by the admission control process. Delay is instead limited by the use of small buffers in the nodes, which should provide packet-scale buffering [65]. The admission-controlled traffic is given strict priority over best-effort traffic, but there is a bandwidth limit. In our simulations, we merely simulated the admission-controlled traffic as being serviced by a queue running at the speed of its bandwidth limit. This is very close to the behavior of a rate-limited priority queue. Simulations are run for 6,000 simulation seconds. The probe loss rate is calculated at the end of each probing period. Moreover, the packet loss probability of the established sessions in the network is also recorded in the simulations for comparisons. We first consider the out-of-band probing scheme, which has two priority queues, one for the high priority accepted data traffic and the other for probe packets. 10 packets of the accepted data traffic and one packet of the probe stream can be buffered at the node. The probing time is 5 seconds.

A robust EMAC should relate the probe loss probability experienced at the receiver to the expected session loss probability. Although this may be true in wireline networks, as shown in previous work, this relationship is very weak in the wireless domain. Figure 5.2 shows the probe loss probability and session loss probability using out-of-band probing in the wireless model. Figure 5.3 shows the probe loss probability and session loss probability using out-of-band probing in the wireline model. Traffic source 1 is used in these simulations. The

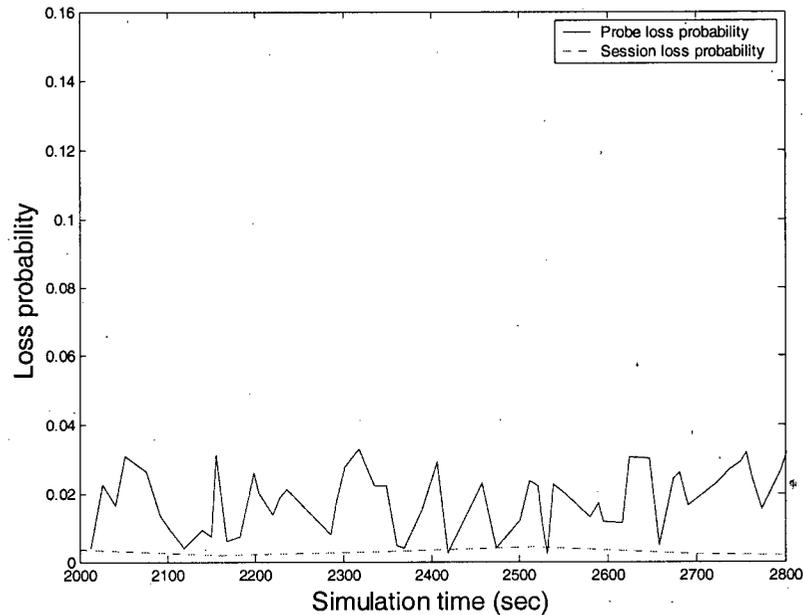


Figure 5.2 Out-of-band probing in the wireline model

offered load is such that the utilizations are 0.56 in both cases. The probe loss rate in the wireless model shows wide fluctuations, ranging from 0.02 to 0.74, although the session data loss rate has little variation. On the other hand, in the wireline case, the probe loss rate is much more constant and may act as a predictor for the session data loss rate. (Note that the scale of the vertical axis in Figure 5.3 is different from that in Figure 5.2.) Similar results have been obtained using traffic source 2. The key reason for the poor performance of out-of-band probing in the wireless domain is the low bandwidth and the burstiness of the data traffic. When the bandwidth is low, the multiplexing level is low and data traffic is active for only short periods of time. As a result, when data traffic is active, the probe packets cannot compete with the data packets, so the probe loss rate varies greatly, even when the utilization of the link is as low as 0.56. We can certainly increase the probing time to improve the performance, but it results in a substantial delay before the host can start sending data and much bandwidth being

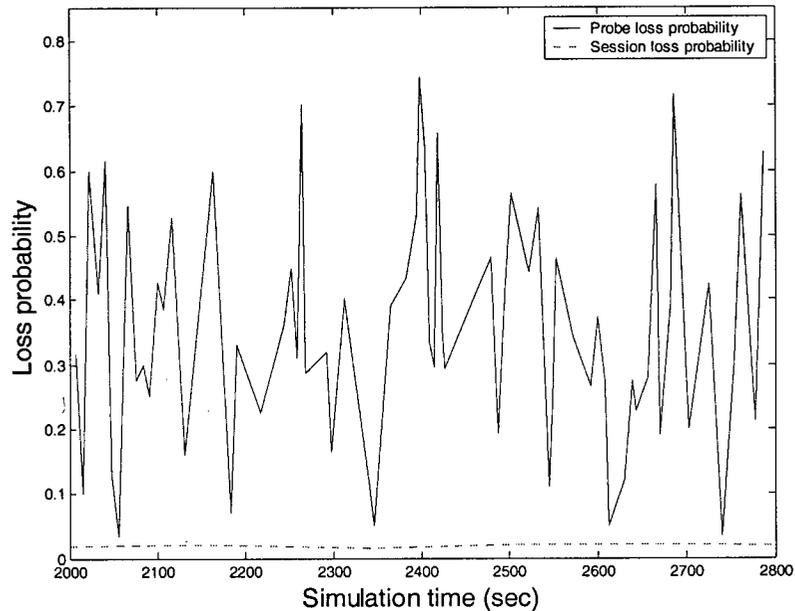


Figure 5.3 Out-of-band probing in the wireless model

wasted, which are not desirable in real networks.

One may think that this problem can be avoided using in-band probing, with the probe packets treated at the same priority level as data. We will study in-band probing in the following. As shown in [19], there is a *thrashing* problem. When so many flows are probing at once that the drop rate is significant, none of the probing flows will be accepted. So, *slow-start* probing is suggested [19], whereby the probing rate slowly ramps up, to detect congestion without unnecessarily creating it. For example, we first probe at rate  $r/16$  for 1 s, where  $r$  is the peak rate of the call; if the loss (or mark) percentage is below a threshold then probe at rate of  $r/8$  for 1 s and the loss rate is checked again. This process is continued for 5 s. Although this scheme can prevent thrashing, the probing result in each shortened step, say 1 s, will not be accurate in predicting the actual session data loss rate, because there is a trade-off between probing time and prediction accuracy, especially in wireless networks. Figures 5.4

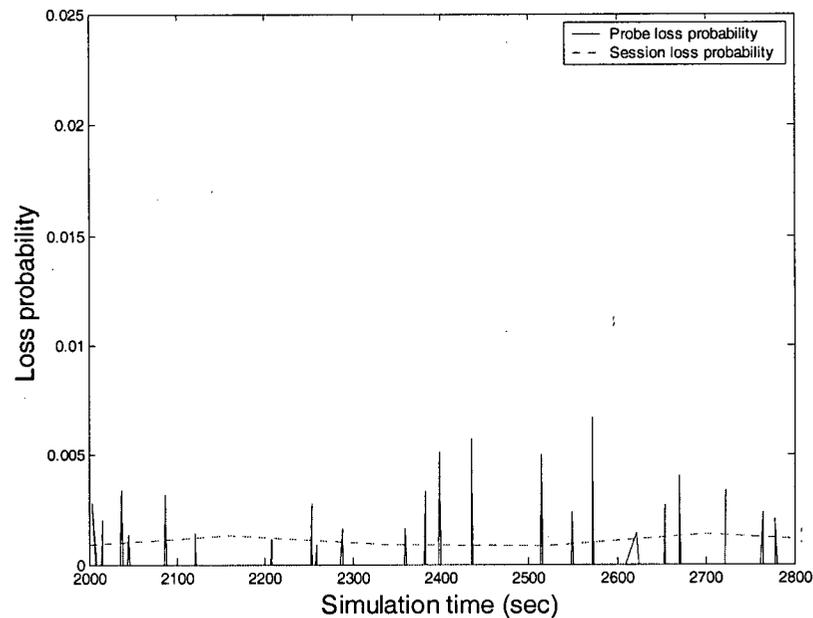


Figure 5.4 In-band probing in the wireline model

and 5.5 show the probe loss probability and session data loss probability using in-band probing with traffic source 1 in the wireless model and wireline model, respectively. In these simulations, the probe is sent with the same priority as data at  $r/4$  rate for 1 s in each case. Again, we see that in the wireless case, the probe loss probability during each 1 s interval varies greatly ranging from 0 to 0.41, thus giving poor predictions of the session data loss rates. (Note again that the scale of the vertical axis in Figure 5.5 is different from that in Figure 5.4.) The results of using traffic source 2 are similar. The poor performance of in-band probing is again due to the low bandwidth and the burstiness of the traffic, and also the short probing period.

### 5.3.2 User Mobility

Since mobile users may change cells a number of times during the lifetime of their

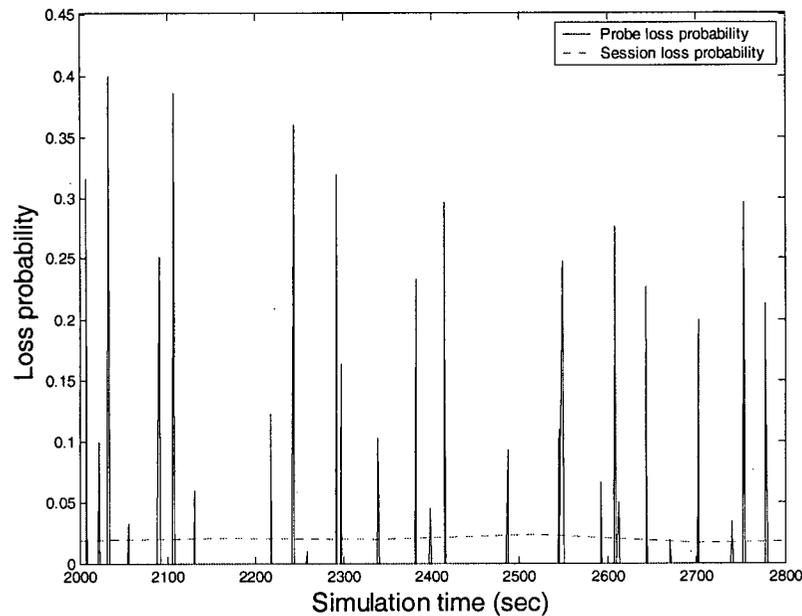


Figure 5.5 In-band probing in the wireless model

calls, availability of wireless network resources at the call setup time does not necessarily guarantee that resources will be available throughout the lifetime of a call. Therefore, the probing result from one cell during the probing phase of a call may not predict accurately the session data loss rate during the lifetime of a call. One possible solution is to probe all the paths that the mobile user will visit during the lifetime. This is however not feasible in practice as probing is done in an end-to-end manner and it is not possible to probe for resource availability in the potential handoff cells in which the mobile user is not currently located. Furthermore, since the mobility of users is usually not known *a priori*, any approach that probes all potential handoff cells will waste a lot of precious wireless bandwidth.

### 5.3.3 High Bit Error Rate

Wireless hosts use radio transmission for communications. This mode of communication is vulnerable to signal degradations due to the propagation environment and interference from other transmissions. BERs on wireless links are found to be about  $10^{-6}$  or worse, compared to  $10^{-12}$  or better for fibre optic links. Moreover, communication over wireless links is often characterized by sporadic high BERs due to multipath fading [7]. The two main classes of techniques employed by reliable link-layer protocols are: forward error correction, and automatic retransmissions of lost packets. In UMTS, retransmission schemes are used at the radio link control/medium access control (RLC/MAC) layers. Since previous EMAC proposals are for wireline networks, only packet losses or congestion marks are considered as acceptance threshold, which is based on the assumption that packets are lost (or marked) because of network congestion. Unfortunately, this is not enough if wireless links are included in the probing path, because packets can be lost for reasons other than congestion and will be retransmitted over the wireless links. We show this by simulations of RLC block transmissions over the air, with random block loss rate 3%. Figure 5.6 shows the sequence numbers of RLC blocks sent at specific time instants with the star dot indicting a lost block that is subsequently retransmitted.

High BER and retransmissions in wireless links complicate the application of EMAC in wireless mobile networks, as they need to be considered in admission decisions, in addition to packet losses and congestion marks.

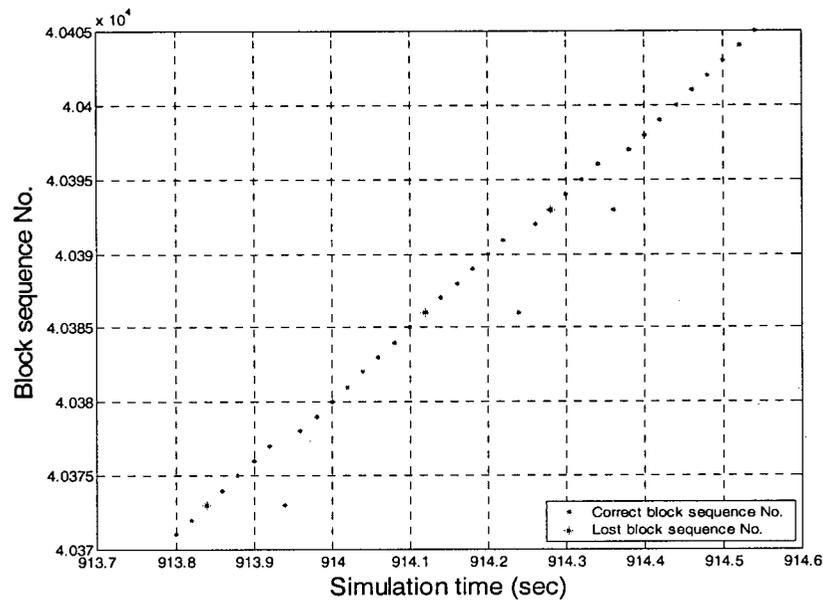


Figure 5.6 Retransmissions of RLC blocks

#### 5.4 Mobile-EMAC for Cellular-to-IP Internetworking

From the analysis and simulation results presented in the last section, we see that the distinct characteristics of cellular mobile networks make EMAC unsuitable for applications in the wireless domain. However, since EMAC has many desirable features in the wireline Internet, especially within the DiffServ framework, it could potentially be widely adopted in the future. Consequently we should have a scheme to internetwork the cellular network to the wireline Internet to add to the potential success of EMAC in the future. In this section, we first review the QoS management functions in UMTS, based on which we propose a mobile-EMAC scheme that incorporates the internetworking function. The performance of the proposed scheme is evaluated by simulations.

### 5.4.1 UMTS QoS Management Functions

Since UMTS is designed to provide multimedia and packet-switched services, Internet and intranet access, entertainment and value-added services, the UMTS standardization process has taken steps to tackle QoS management functions in the 3rd generation partnership project (3GPP) [2]. QoS provisioning in UMTS is based on the traffic classification model with four main traffic classes: Conversational, Streaming, Interactive and Background classes. The main distinguishing factor between these classes is the delay sensitiveness of the traffic: Conversational class has the most stringent delay requirement, whereas Background class is the most delay insensitive class.

In addition to classifying the traffic, the network needs to have means to provide and maintain the QoS for each admitted user traffic stream. UMTS QoS management functions can be logically divided into control and user plane functions. Call admission control addressed in this study belongs to the control plane functions described below.

The control plane incorporates the following four main entities. The service manager co-ordinates the functions of the control plane for establishing, modifying and maintaining the service it is responsible for. The translation function converts between the internal service primitives for UMTS bearer service control and the various protocols for service control of external networks that are interconnected to the UMTS network, including conversion between UMTS BS attributes and QoS attributes of the external networks' service control protocols. Admission/capability control maintains information about all available resources of a network entity and resources allocated to all UMTS bearer services, determines for each

UMTS bearer service request or modification whether the required resources can be provided by this entity, and reserves these resources if they are allocated to the UMTS bearer service. Subscription control checks the administrative rights of the UMTS bearer service users to use the requested services with the specified QoS attributes. The QoS management functions for controlling the UMTS bearer service are shown in Figure 5.7. These functions support the establishment and modification of UMTS bearer services by signaling/negotiation with the UMTS external services and by the establishment or modification of appropriate UMTS internal services with the required characteristics.

### 5.4.2 Mobile-EMAC Scheme

Since UMTS already incorporates admission control and capacity allocation functions

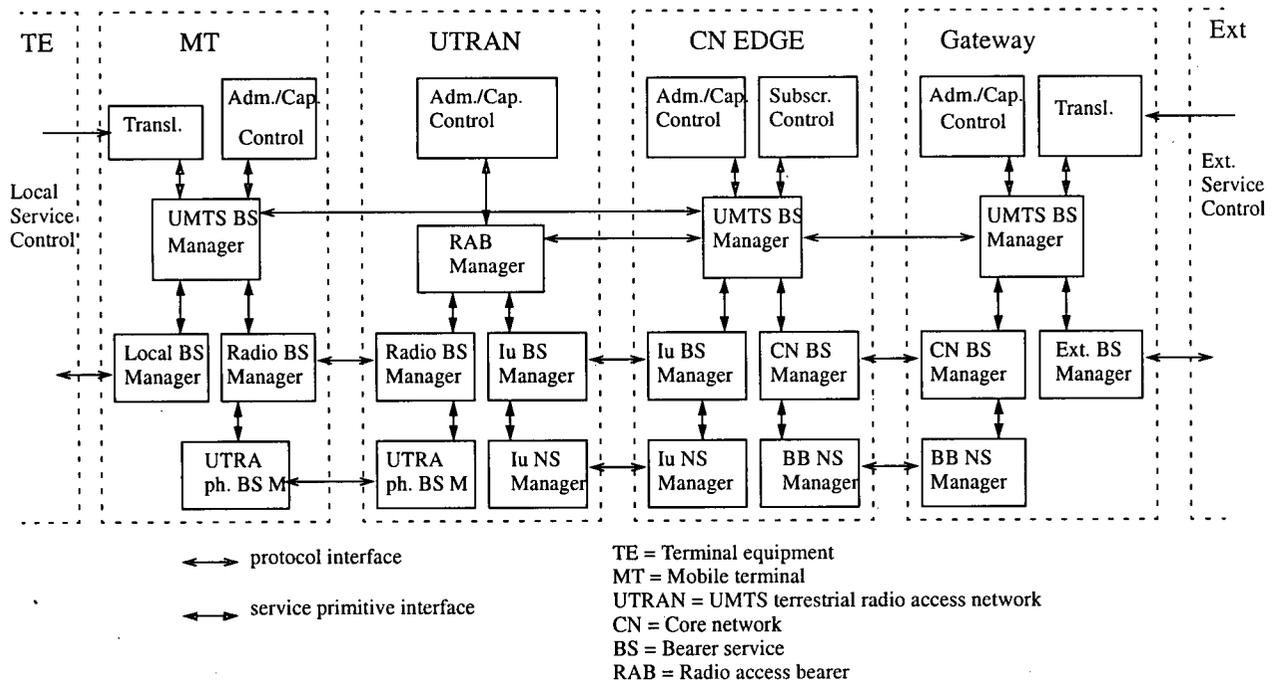


Figure 5.7 QoS management functions for the UMTS bearer service in control plane

in its control plane that will be optimized for the wireless environment, information for call admission decision is readily available in the network elements. Thus the use of probing within the UMTS segment is unnecessary. Moreover, the UMTS control plane specifically includes translation functions that can be used to facilitate internetworking between UMTS and the various protocols for service control of external networks. Therefore, in our proposed M-EMAC, we terminate the EMAC probing for the wireline Internet at the UMTS gateway that serves as a pseudo-end-point for the EMAC protocol, as shown in Figure 5.8. We assume the majority of the traffic is sent from a remote host to the mobile host. In this case the gateway receives the probing packets and translates the QoS parameters inside the probing packets into UMTS QoS parameters. Then the admission/capacity control functions in UMTS check whether or not such a call request can be admitted and return the result using UMTS signaling. After the gateway gets both the checking result within UMTS and the probing result from the Internet, an admission decision can be made based on these results and the end hosts are informed accordingly. Figure 5.9 shows this procedure. In this internetworking

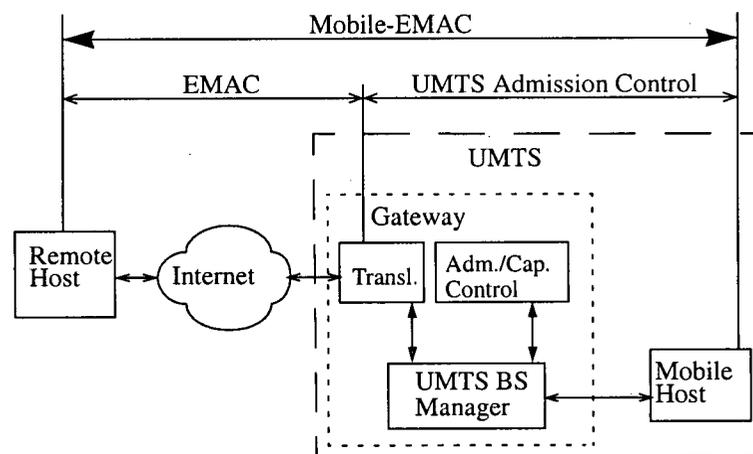


Figure 5.8 Mobile-EMAC

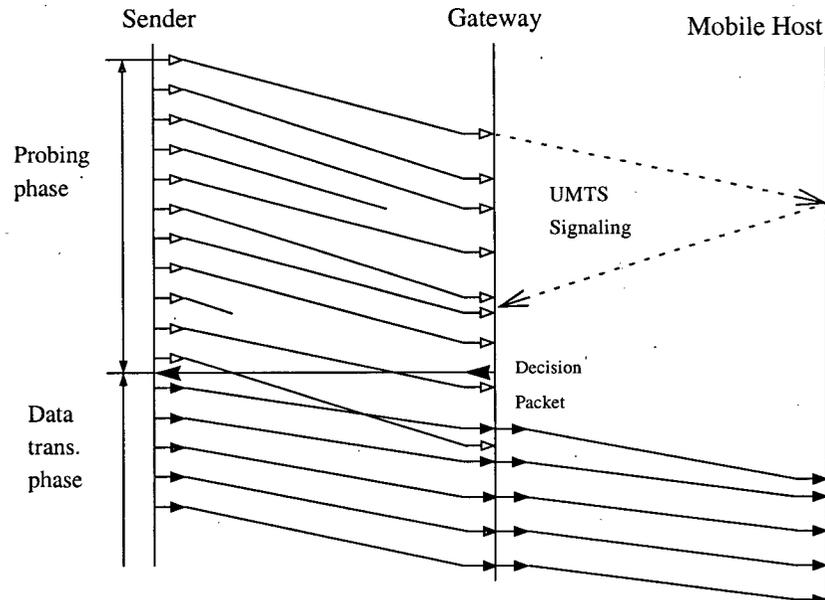


Figure 5.9 Mobile-EMAC procedure

approach, the probing packets need not go through the UMTS network and we can avoid the problems described in Section 5.3.

### 5.4.3 Performance Evaluation

In this subsection, we compare the performance of EMAC in wireless domain with that of the proposed M-EMAC. The simulation models include the EDGE model and the 10 Mbps wireline model described in Section 5.3. The traffic is sent from a server to a mobile via the 10 Mbps wireline network and the wireless EDGE network. There is a gateway between the wireless and wireline networks, which has the functions described in the last subsection. The radio link assumptions are the same as in Section 5.3, i.e., a fixed modulation and coding scheme. The traffic sources are also the same as before. In the M-EMAC scheme, the tradi-

tional measurement-based admission control (MBAC) [43] is used in the wireless network. As shown in [20], although various MBAC algorithms in the literature are derived from diverse motivations and theories, they all produce essentially the same performance. So, we use a simple measured sum (MS) algorithm [20] in the wireless part of the M-EMAC scheme. In EMAC schemes, we test the out-of-band probing in EMAC with acceptance thresholds  $\epsilon = 0.05, 0.09, 0.15, 0.25, 0.4$ , and the in-band probing in EMAC with  $\epsilon = 0.01, 0.03, 0.05, 0.07, 0.09$ .

We evaluate EMAC and the proposed M-EMAC algorithms according to how well they are able to achieve the highest possible utilization for a given level of service failure probability. We compare the performance *frontier* or *loss-load curve* achieved by each algorithm. The loss-load curve depicts the rate of packet losses that occur at a given level of wireless channel utilization, as experienced by the mobile host. Figures 5.10 and 5.11 show loss-load curves of EMAC and M-EMAC. For a given EMAC algorithm (in-band probing or out-of-band probing) each point shown on the curve reflects the loss probability and utilization produced by a different  $\epsilon$  value. For the M-EMAC scheme, we test the utilization with different loss rates of 0.0009, 0.003, 0.01, 0.03 and 0.09. Simulations are run for 6000 simulation seconds. Data collected during the first 1500 seconds are discarded. The plot data are obtained by averaging over 10 simulation runs with different random seeds. Traffic source 1 is used in these simulations. Other simulation parameters are the same as presented in the previous section.

In Figure 5.10, where the probing time is 5 seconds, the frontiers of EMAC with in-

band and out-of-band probing seem to be fairly close to each other. However, for a given level of utilization the loss rates achieved by EMAC (both in-band probing and out-of-band probing) are much higher than those achieved by M-EMAC. The reason is that the probe loss rates do not give very accurate predictions of the session data loss rates in the wireless domain, as explained in Section 5.3. EMAC could presumably achieve better performance by probing for longer time periods. Figure 5.11 shows the loss-load curves achieved by EMAC with 20 second probes. We can see that out-of-band probing for 20 seconds can achieve a better performance than probing for 5 seconds (but still worse than that of M-EMAC), since a longer probing time yields probe loss rates that are closer to the session loss rates. However, while a longer time for in-band probing leads to decreased packet drop rates, they can cause the utilization to be decreased substantially. Since the probe and data packets have the same priority, longer probing time means that more bandwidth is consumed by probe packets. We can get similar results using traffic source 2. These results show that the proposed M-EMAC outperforms EMAC in the mobile/wireless domain.

## 5.5 Summary

In this chapter, we have studied the performance of EMAC in cellular mobile networks. Simulation and analysis results show that EMAC may not be suitable for cellular mobile environments due to the scarce bandwidth, user mobility and high bit error rate. On the other hand, cellular mobile network such as UMTS already includes call control and capacity allocation functions optimized for its environment. This motivates the M-EMAC scheme proposed in this study for cellular-to-IP internetworking. In the M-EMAC scheme, the gateway makes admission decisions based on the results of EMAC probing on the

wireline side and the admission control on the wireless side. Simulation results show that M-EMAC outperforms EMAC in the cellular mobile domain in terms of the end-to-end packet loss rate at any given level of channel utilization.

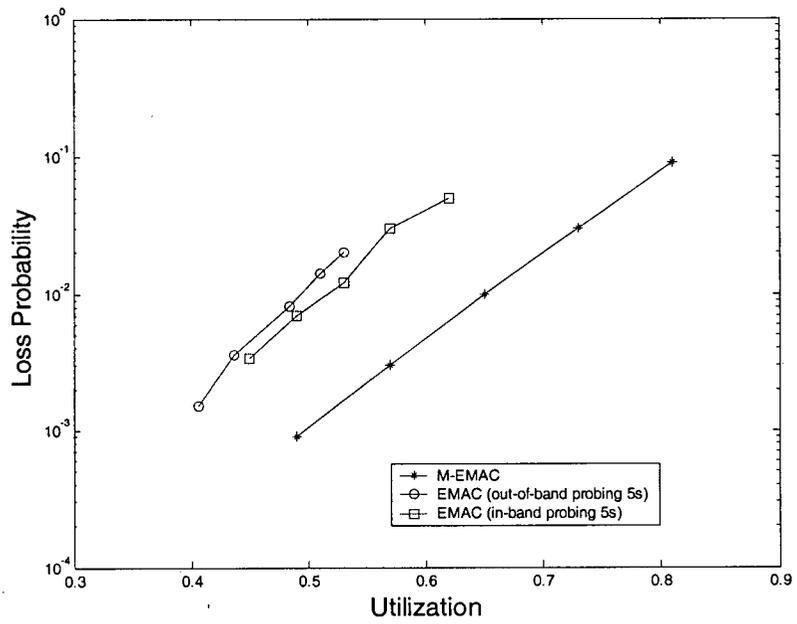


Figure 5.10 Loss-load curves of M-EMAC and EMAC (probing 5 s)

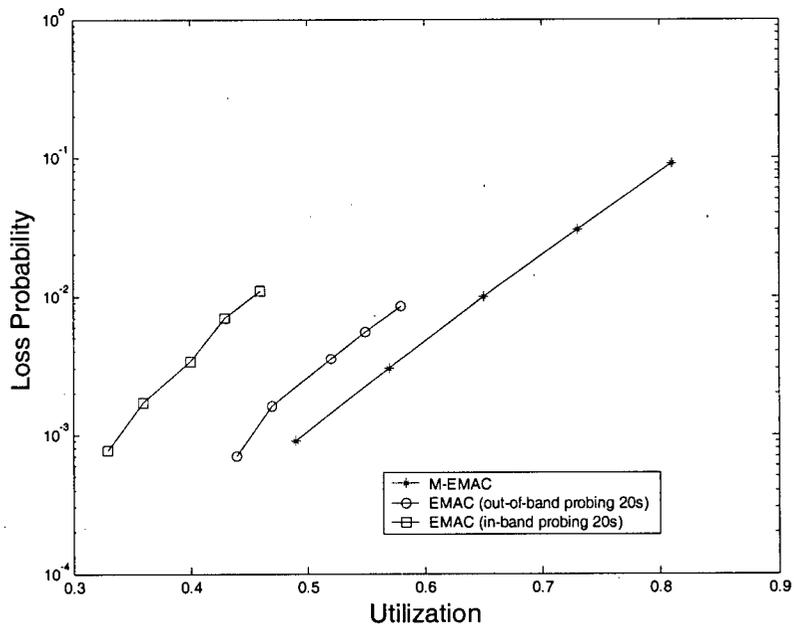


Figure 5.11 Loss-load curves of M-EMAC and EMAC (probing 20 s)

## Chapter 6 Conclusions

We conclude this thesis with a summary of our contributions and suggestions for future work.

### 6.1 Summary

In Chapter 2, we proposed call admission control and bandwidth reservation schemes for non-adaptive traffic in cellular mobile networks, based on assumptions more realistic than in existing proposals. The sequences of events of new call admission, handoffs and call termination are modeled by stationary  $m$ th order Markov sources. We derived probabilistic predictions of next events based on the mobility history of users, using an algorithm motivated by optimal data compression. Based on the mobility prediction of where and when the mobile will handoff to the next cell, call admission control and bandwidth reservation schemes were developed. The performance of the proposed schemes has been studied using computer simulations. Handoff dropping probabilities can be kept below a target value in the proposed call admission control and bandwidth reservation schemes. It is also shown that our schemes can achieve a better balance of guaranteeing handoff dropping probability while maximizing resource utilization, and they outperform the static reservation scheme.

In Chapter 3, we presented a framework combining QoS provisioning and mobility management. In this framework, we use all available information, both during calls and between calls to build the mobility trie, which is used in mobility prediction. Networks can use this mobility prediction scheme in both paging mobiles and making admission decisions. A new path-based MM scheme in the combined framework was proposed. Numerical results

show that the new scheme has performance gain over the original one, in that the new scheme can reduce both location update and paging costs. A new CAC scheme in our framework was proposed in the framework. The proposed CAC scheme can predict where the mobile will handoff using the common mobility prediction scheme and when the handoff will occur using positioning technology. Based on mobility predictions, bandwidth can be reserved more efficiently. Simulation results show that the proposed scheme meets our design goal and outperforms some existing schemes.

In Chapter 4, we proposed a QoS provisioning scheme for adaptive multimedia in cellular mobile networks. The rapid growth in the number of states and the difficulty to estimate the state transition probabilities in practical cellular systems motivate us to choose a model-free reinforcement learning solution to solve this problem. Three QoS constraints, handoff dropping probability, average allocated bandwidth and intra-class fairness are considered in this scheme. The algorithm can control the adaptation frequency effectively by formulating the cost of bandwidth adaptation in the model. Trading off action space with state space was proposed to solve the large action space problem in QoS provisioning. The performance of the proposed scheme was demonstrated by simulations. The algorithm works well under a variety of traffic conditions with reasonable computation complexity.

Finally, we studied the performance of EMAC in cellular mobile networks in Chapter 5. Although EMAC has many desirable features in wireline networks, simulation and analysis results show that EMAC may not be suitable in a cellular mobile environment due to the scarce bandwidth, user mobility, and high bit error rate. Since EMAC has many desirable

features in the wireline Internet, especially within the DiffServ framework, it could potentially be widely adopted in the future. We proposed an M-EMAC scheme to interconnect the cellular network to the wireline Internet to add to the potential success of EMAC in the future. In the M-EMAC scheme, the gateway makes admission decisions based on the results of EMAC probing on the wireline side and the admission control on the wireless side. Simulation results show that the loss rates achieved by EMAC are much higher than those achieved by the proposed M-EMAC for a given level of utilization. These results demonstrate that M-EMAC outperforms EMAC in cellular mobile networks.

## 6.2 Further Work

In the course of the investigations reported in this thesis, a number of interesting problems have been discovered which merit further research.

- **More Realistic Traffic and Mobility Models:** Traffic and mobility models are essential for solving QoS provisioning problems in future cellular mobile networks. Although there are some traffic and mobility models available in the literature, new applications, such as multimedia message services (MMS), gaming and positioning applications, are emerging very fast. Few of the existing models are general enough to provide good approximations for these new applications. Further work is required in this area.
- **Vertical Handoff:** Future wireless mobile networks will be heterogeneous. Simultaneously there will be a proliferation of different wireless access technologies - from those providing global coverage, such as cellular mobile networks, to those provid-

ing wireless hotspots such as wireless local area network (WLAN). As a result, mobile communications will be conducted over heterogeneous wireless networks. Continuous connectivity and QoS provisioning in these networks remain open issues.

- **Other QoS Provisioning Schemes in the Internet:** Many other schemes to provide QoS in the Internet have been proposed, including new scheduling algorithms and new CAC schemes. To guarantee the end-to-end QoS of cellular mobile users, there may be some internetworking issues for mobile users accessing the wireless Internet. Finding an efficient way to internetwork cellular mobile network to the Internet is a very interesting problem.
- **User Profile:** Some schemes in this thesis predict user mobility based on mobile users' histories, positions, velocities, etc. This information has to be stored in the user profile, which is very important to provide personalized services in future cellular mobile networks. An efficient way of collecting, storing, updating and disseminating the user profile information is crucial.

## Bibliography

- [1] 3GPP, "RRC protocol specification," TS25.331 version 3.12.0, Sept. 2002.
- [2] 3GPP, "Technical Specification Group Services and System Aspects, QoS Concept and Architecture," TS 23.107 version 3.9.0, Sept. 2002.
- [3] EIA/TIA IS-41.3 (Revision B), *Cellular Radio-Telecommunications Intersystem Operations*, July 1991.
- [4] T1P1, *Evaluation Sheet for the Enhanced Observed Time Difference (E-OTD) Method*, Doc. T1P1.5/98-021r8, Aug. 1998.
- [5] I. F. Akyildiz, J. Ho, and Y.-B. Lin, "Movement-based location update and selective paging for PCS networks," *IEEE/ACM Trans. Networking*, vol. 4, no. 4, pp. 629-638, Aug. 1996.
- [6] E. Altman, *Constrained Markov decision process*, Chapman and Hall, London, 1999.
- [7] H. Balakrishnan, V.N. Padmanabhan, S. Seshan and R.H. Katz, "A comparison of mechanisms for improving TCP performance over wireless links", *IEEE/ACM Trans. Networking*, vol. 5, no. 6, pp. 756-769, Dec. 1997.
- [8] A. Bar-Noy, I. Kessler, and M. Sidi, "Mobile users: to update or not to update?," *Wireless Networks*, vol. 1, no.2, pp. 175-185, July 1995.
- [9] T. C. Bell, J. C. Cleary and I. H. Witten, *Text Compression*, Prentice-Hall Advanced Reference Series, Prentice-Hall, Englewood Cliffs, NJ, 1990.
- [10] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-dynamic programming*, Athena Scientific, Belmont, MA, 1996.
- [11] F. J. Beutler and K. W. Ross, "Optimal policies for controlled Markov chains with a constraint," *J. Math. Anal. Appl.*, vol. 112, pp. 236-252, 1985.
- [12] F. J. Beutler and K. W. Ross, "Time-average optimal constrained semi-Markov decision processes," *Adv. Appl. Prob.*, vol. 18, pp. 341-359, 1986.

- [13] A. Bhattacharya and S. K. Das, "LeZi-update: An information-theoretic framework for personal mobility tracking in PCS networks," *Wireless Networks*, vol. 8, no. 2, pp. 121-135, May 2002.
- [14] G. Bianchi, A. Capone, and C. Petrioli, "Throughput analysis of end-to-end measurement-based admission control in IP", in *Proc. IEEE INFOCOM'00*, vol. 3, Tel Aviv, Israel, Mar. 2000, pp. 1461 -1470.
- [15] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss, "An architecture for differentiated services," *IETF RFC 2475*, Dec. 1998.
- [16] V. Bolotin, "Modeling call holding time distributions for CCS network design and performance analysis", *IEEE J. Sel. Areas in Commun.*, vol. 12, no. 3, pp. 433-438, Apr. 1994.
- [17] R. Board and L. Pitt, "On the necessity of occam algorithms," in *Proceedings of the 22nd Annual ACM Symposium on Theory of Computation*, New York, NY, May 1990.
- [18] R. Braden, D. Clark, and S. Shenker, "Integrated services in the internet architecture: an overview," *IETF RFC1633*, June 1994.
- [19] L. Breslau, E. W. Knightly, S. Shenker, I. Stoica and H. Zhang, "Endpoint admission control: architectural issues and performance", in *Proc. ACM SIGCOMM'00*, Stockholm, Sweden, Aug. 2000, pp. 57-69.
- [20] L. Breslau, S. Jamin, and S. Shenker, "Comments on the performance of measurement-based admission control algorithms", in *Proc. IEEE INFOCOM'00*, vol. 3, Tel Aviv, Israel, Mar. 2000, pp. 1233 -1242.
- [21] S. Bunton, G. Borriello, "Practical dictionary management for hardware data compression," *Commun. ACM*, vol. 35, no. 1, pp.95-104, Jan. 1992.
- [22] C. Cetinkaya and E. Knightly, "Egress admission control", in *Proc. IEEE INFOCOM'00*, vol. 3, Tel Aviv, Israel, Mar. 2000, pp. 1471-1480.
- [23] C. Chao and W. Chen, "Connection admission control for mobile multiple-class per-

- sonal communications networks," *IEEE J. Select. Areas Commun.*, vol. 15, No. 8, pp. 1618-1626, Oct. 1997.
- [24] S. Choi and K. G. Kin, "Adaptive bandwidth reservation and admission control in QoS-sensitive cellular networks," in *IEEE Trans. Parallel and Distributed Systems*, vol. 13, no. 9, pp. 882-897, Sept. 2002.
- [25] C. Chou and K. G. Shin, "Analysis of combined adaptive bandwidth allocation and admission control in wireless networks," in *Proc. IEEE INFOCOM'02*, vol. 2, New York, NY, June 2002, pp. 676-684.
- [26] T. K. Das, A. Gosavi, S. Mahadevan and N. Marchallick, "Solving semi-markov decision problems using average reward reinforcement learning," *Management Science*, vol. 45, no. 4, pp. 560-574, 1999.
- [27] C. Darken, J. Chang, and J. Moody, "Learning rate schedules for faster stochastic gradient search," in *Proc. IEEE Workshop on Neural Networks for Signal Processing*, Helsingoer, Denmark, Sept. 1992.
- [28] V. Elek, G. Karlsson, and R. Ronngren, "Admission control based on end-to-end measurements", in *Proc. IEEE INFOCOM'00*, vol. 2, Tel Aviv, Israel, Mar. 2000, pp. 623-630.
- [29] A. I. Elwalid and D. Mitra, "Effective bandwidth of general Markovian traffic sources and admission control of high-speed networks," *IEEE/ACM Trans. Networking*, vol. 1, no. 3, pp. 329-343, June 1993.
- [30] B. T. Fang, "Simple solutions for hyperbolic and related position fixes," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 26, no. 5, pp. 748-753, Sept. 1990.
- [31] Y. Fang and I. Chlamtac, "Teletraffic analysis and mobility modelling of PCS networks," *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1062-1072, July 1999.
- [32] R. L. Freeman, *Telecommunication System Engineering*, 3rd ed., New York: Wiley, 1996.

- [33] A. Furuskar, M. Frodigh, H. Olofsson, and J. Skold, "System performance of EDGE, a proposal for enhanced data rate in existing digital cellular systems", in *Proc. IEEE VTC'98*, vol. 2, Ottawa, Canada, May, 1998, pp. 1284 -1289.
- [34] Z. Gabor, Z. Kalmar, and C. Szepesvari, "Multi-criteria reinforcement learning," In *Proc. Int'l Conf. Machine Learning*, Madison, WI, July 1998.
- [35] S. Ganguly and B. Nath, "QoS provisioning for adaptive services with degradation in cellular network," in *Proc. IEEE WCNC'03*, New Orleans, LA, Mar. 2003.
- [36] R. Gibbens and F. Kelly, "Distributed connection acceptance control for a connection-less network", in *Proc. ITC'99*, Edinburgh, UK, June 1999.
- [37] A. Gosavi, *An algorithm for solving semi-Markov decision problems using reinforcement learning: convergence analysis and numerical results*, Ph.D. Dissertation, University of South Florida, 1999.
- [38] A. Gosavi, N. Bandla and T. K. Das, "A reinforcement learning approach to airline seat allocation for multiple fare classes with overbooking," *IIE Transaction*, vol. 34, pp. 729-742, 2002.
- [39] R. Guerin, H. Ahmadi, and M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks," *IEEE J. Select. Areas Commun.*, vol. 9, no. 7, pp. 968-981, Sept. 1991.
- [40] M. Hellebrandt, R. Mathar, and M. Scheibenbogen, "Estimating position and velocity of mobiles in a cellular radio network," *IEEE Trans. Veh. Technol.*, vol. 46, pp.65-71, Feb. 1997.
- [41] D. Hong and S. S. Rappaport, "Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized handoff procedure," *IEEE Trans. Veh. Technol.*, vol. VT-35, no. 3, pp. 77-92, 1986.
- [42] F. Kielly, P. Key, and S. Zachary, "Distributed admission control," *IEEE J. Select. Areas Commun.*, vol. 18, no. 12, pp 2617-2628, Dec. 2000.

- [43] S. Jamin, P. B. Danzig, S. J. Shenker, L. Zhang, "A measurement-based admission control algorithm for integrated services packet networks", *IEEE/ACM Trans. Networking*, vol. 5, no. 1, pp. 56-70, Feb. 1997.
- [44] C. Jedrzycki and V. C. M. Leung, "Probability distributions of channel holding time in cellular telephony systems," in *Proc. IEEE VTC'96*, vol. 1, Atlanta, GA, May 1996, pp. 247-251.
- [45] T. Kwon, Y. Choi, C. Bisdikian and M. Naghshineh, "QoS provisioning in wireless/mobile multimedia networks using an adaptive framework," *Wireless Networks*, vol. 9, no. 1, pp. 51-59, 2003.
- [46] G. G. Langdon, "A note on Ziv-Lempel model for compressing individual sequences," *IEEE Trans. Inf. Theory*, vol. 29, no. 2, pp. 284-287, Mar. 1983.
- [47] D. A. Levine, I. F. Akyildiz, and M. Naghshineh, "A resource estimation and call admission algorithm for wireless multimedia networks using the shadow cluster concept," *IEEE/ACM Trans. Networking*, vol. 5, no. 1, pp. 1-12, Feb. 1997.
- [48] B. Liang and Z. Haas, "Predictive distance-based mobility management for PCS networks," in *Proc. IEEE INFOCOM'99*, vol. 3, New York, NY, Mar. 1999, pp. 1377-1384.
- [49] Y. B. Lin, S. Mohan, and A. Noerpel, "Queueing priority channel assignment strategies for handoff and initial access for a PCS network," *IEEE Trans. Veh. Technol.*, vol. 43, no.3, pp. 704-712, Aug. 1994.
- [50] Y. B. Lin, A. Noerpel, and D. Harasty, "The sub-rating channel assignment strategy for PCS hand-offs," *IEEE Trans. Veh. Technol.*, vol. 45, no. 1, pp. 122-130, Feb. 1996.
- [51] S. Lu and V. Bharghavan, "Adaptive resource management algorithms for indoor mobile computing environment," in *Proc. ACM SIGCOMM'96*, Palo Alto, CA, Sept. 1996, pp. 231-242.
- [52] S. Mahadevan, "Average reward reinforcement learning: Foundations, algorithms, and empirical results," *Machine Learning*, vol. 22, pp. 159-196, 1996.

- [53] D. Mitra, M. I. Reiman and J. Wang, "Robust dynamic admission control for unified cell and call QoS in statistical multiplexers," *IEEE J. Select. Areas Commun.*, vol. 16, no. 5, pp. 692-707, June 1998.
- [54] M. Mouly and M.-B. Pautet, *The GSM System for Mobile Communications*, Palaiseau, France, 1992.
- [55] M. Naghshineh and M. Schwartz, "Distributed call admission control in mobile/wireless networks," *IEEE J. Select. Areas Commun.*, vol. 14, no. 4, pp. 711-717, May 1996.
- [56] M. Naghshineh and A. S. Acampora, "QoS provisioning in micro-cellular networks supporting multiple classes of traffic," *Wireless Networks*, vol. 2, no. 3, pp. 195-203, 1996.
- [57] C. Oliveira, J. B. Kim, and T. Suda, "An adaptive bandwidth reservation scheme for high-speed multimedia wireless networks," *IEEE J. Select. Areas Commun.*, vol. 16, no. 6, pp. 858-874, Aug. 1998.
- [58] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, 3rd ed., 1991.
- [59] G. P. Pollini and C.-L. I, "A profile-based location strategy and its performance," *IEEE J. Select. Areas Commun.*, vol. 15, no. 8, pp. 1415-1425, Oct. 1997.
- [60] E. C. Posner and R. Guerin, "Traffic policies in cellular radio that minimize blocking of handoff calls," in *Proc. 11th Teletraffic Cong. (ITC-11)*, Kyoto, Japan, Sept. 1985.
- [61] M. L. Puterman, *Markov Decision Processes*, Wiley Interscience, New York, USA, 1994.
- [62] R. Ramjee, D. Towsley, and R. Nagarajan, "On optimal call admission control in cellular networks," *Wireless Networks*, vol. 3, no. 1, pp. 29-41, 1997.
- [63] T. S. Rappaport, *Wireless Communication: Principles and Practice*, Englewood Cliffs, NJ: Prentice Hall, 1996.
- [64] J. H. Reed, K. J. Krizman, B. D. Woerner, and T. S. Rappaport, "An overview of the challenges and progress in meeting the E-911 requirement for location service," *IEEE*

- Commun. Mag.*, vol. 36, no. 4, pp. 30-37, Apr. 1998.
- [65] J. Roberts *et al.*, "Broadband network teletraffic, Final report of action COST 242", Springer, 1996.
- [66] S. K. Sen, A. Bhattacharya, and S. K. Das, "A selective location update strategy for PCS users," *Wireless Networks*, vol. 5, no.5, pp. 313-326, Oct. 1999.
- [67] D. Sheinwald, "On the Ziv-Lempel proof and related topics," *Proc. IEEE*, vol. 82, no. 6, pp. 866-871, June 1994.
- [68] D. D. Sleator and R. E. Tarjan, "Amortized efficiency of list update and paging rules," *Commun. ACM* 28, vol. 2, pp. 202-208, Feb. 1985.
- [69] J. A. Storer, *Data Compression Methods and Theory*, Computer Science Press, Rockville, MD, 1998.
- [70] S. Tabbane, "An alternative strategy for location tracking," *IEEE J. Select. Areas Commun.*, vol. 13, no. 5, pp. 880-892, June 1995.
- [71] A. K. Talukdar, B. R. Badrinath, and A. Acharya, "Rate adaptive schemes in networks with mobile hosts," in *Proc. ACM/IEEE MOBICOM'98*, Dallas, TX, Oct. 1998, pp. 169-180.
- [72] A. K. Talukdar, B. R. Badrinath and A. Acharya, "Integrated services packet networks with mobile host: Architecture and performance," *Wireless Networks*, vol. 5, no. 2, pp. 111-124. 1999.
- [73] D. Taubman and A. Zakhor, "A common framework for rate and distortion based scaling of highly scalable compressed video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 4, pp. 329-354, Aug. 1996.
- [74] H. Tong and T. X. Brown, "Adaptive call admission control under quality of service constraints: a reinforcement learning solution," *IEEE J. Select. Areas Commun.*, vol. 18, no. 2, pp. 209-221, Feb. 2000.
- [75] A. J. Viterbi, *CDMA: Principle of Spread Spectrum Communication*, Addison-Wesley,

Reading, MA, 1995.

- [76] J. S. Vitter and P. Krishnan, "Optimal prefetching via data compression," *J. ACM*, vol. 43, no.5, pp. 771-793, Sept. 1996.
- [77] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, pp. 279-292, 1992.
- [78] V. W. S. Wong and V. C. M. Leung, "Location management for next-generation personal communications networks," *IEEE Networks*, vol. 14, no. 5, pp. 18-24, Oct. 2000.
- [79] V. W. S. Wong and V. C. M. Leung, "An adaptive distance-based location update algorithm for next-generation PCS networks," *IEEE J. Select. Areas Commun.*, vol. 19, no. 10, pp. 1924-1952, Oct. 2001.
- [80] I. H. Witten, R. M., Neal and J. G. Cleary, "Arithmetic coding for data compression," *Commun. ACM*, vol. 30, no. 6, pp. 520-540, June 1987.
- [81] D. Wu, Y. T. Hou, and Y. Q. Zhang, "Scalable video coding and transport over broadband wireless networks," *Proc. IEEE*, vol. 89, no.1, pp. 6-20, Jan. 2001.
- [82] Y. Xiao, P. Chen, and Y. Wang, "Optimal admission control for multi-class of wireless adaptive multimedia services," *IEICE Trans. Commun.*, vol. E84-B, no. 4, pp. 795-804, Apr. 2001.
- [83] F. Yu and V. C. M. Leung, "Mobility-based predictive call admission control and bandwidth reservation in wireless cellular networks," in *Proc. IEEE INFOCOM'01*, vol. 1, Anchorage, Alaska, Apr. 2001, pp. 518-526.
- [84] F. Yu and V. C. M. Leung, "A framework of combining connection admission control and mobility management in wireless cellular networks," in *Proc. IEEE ICC'01*, vol. 7, Helsinki, Finland, June 2001, pp. 2286-2290.
- [85] F. Yu and V. C. M. Leung, "Connection admission control for PCS-to-IP interworking," in *Proc. IEEE PIMRC'01*, vol. 1, San Diego, CA, Sept. 2001, pp. B70-B74.
- [86] F. Yu and V. C. M. Leung, "Connection admission control for PCS-to-IP interworking,"

- Kluwer Int'l J. Wireless Information Networks*, vol. 9, no. 1, pp.1-12, Jan. 2002.
- [87] F. Yu and V. C. M. Leung, "Mobility-based predictive call admission control and bandwidth reservation in wireless cellular networks," *Elsevier Computer Networks*, vol. 38, no. 5, pp. 577-589, Apr. 2002.
- [88] F. Yu and V. W. S. Wong and V. C. M. Leung, "Performance enhancements of combining QoS provisioning and location management in wireless cellular networks," in *Proc. IEEE Globecom'02*, vol. 1, Nov. 2002, Taipei, Taiwan, pp. 956-960.
- [89] F. Yu and V. W. S. Wong and V.C.M. Leung, "Performance enhancements of combining QoS provisioning and location management in wireless cellular networks," accepted and to appear in *IEEE Trans. Wireless Communications*, Sept. 2003.
- [90] F. Yu, V.W.S. Wong and V.C.M. Leung, "Reinforcement Learning For Call Admission Control and Bandwidth Adaptation in Mobile Multimedia Networks," to be presented at ICICS-PCM'03, Singapore, Dec. 2003
- [91] F. Yu, V.W.S. Wong and V.C.M. Leung, "Efficient QoS Provisioning for Adaptive Multimedia in Mobile Communication Network by Reinforcement Learning," to be presented at ACM/SPIE Conference on Multimedia Computing and Networking (MMCN), Santa Clara, CA, Jan. 2004.
- [92] G. V. Zaruba, I. Chlamtac and S. K. Das, "A prioritised real-time wireless call degradation framework for optimal call mix selection," *Mobile Networks and Applications*, vol. 7, no. 2, pp. 143-151, 2002.
- [93] H. Zhang and D. Ferrari, "Rate-controlled service disciplines", *Journal of High Speed Networks*, vol. 3, no. 4, pp. 389-412, 1994.
- [94] W. Zhuang, B. Bensaou, and K. C. Chua, "Adaptive quality of service handoff priority scheme for mobile multimedia networks," *IEEE Trans. Veh. Technol.*, vol. 49, no. 2, pp. 494-505, Mar. 2000.
- [95] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *IEEE Trans. Inf. Theory*, vol. 24, no. 5, pp. 530-536, Sept. 1978.