International Journal of Wireless Information Networks, Vol. 10, No. 2, April 2003 (© 2003)

A Call-Admission Control (CAC) Algorithm for Providing Guaranteed QoS in Cellular Networks

Satya Kovvuri,[1] Vijoy Pandey,[2] Dipak Ghosal,[2] Biswanath Mukherjee,[2] and Dilip Sarkar,[1,3]

Future broadband wireless access systems are expected to integrate various classes of mobile terminals (MTs), each class with a different type of quality of service (QoS) requirement. When the load on a wireless network is high, the guarantee of QoS for each class of MTs is a challenging task. This study considers two classes of MTs—*profiled* MTs and *nonprofiled* or *regular* MTs. It is assumed that profiled users require a guaranteed QoS. The measure of QoS is the probability of forced termination of a call that was allowed to access the network. Two previous handoff prioritization schemes—(i) *prerequest scheme* and (ii) *guard channel scheme*—decrease handoff failure (and hence forced termination). In this work, we compare and contrast both the schemes through extensive simulation and we find that neither guard channel nor channel prerequest scheme can guarantee a desired level of QoS for the profiled MTs. We then propose a novel call-admission control (CAC) algorithm that can maintain any desired level of QoS, while the successful call completion rate is very high. In the proposed algorithm, the new call arrival rate is estimated continuously, and when the estimated arrival rate is higher than a predetermined level, some new calls are blocked irrespective of the availability of channels. The objective of this new call preblocking is to maintain a cell's observed new call arrival rate at no more than the predetermined rate. We show that the proposed method can guarantee any desired level of QoS for profiled users.

KEY WORDS: Call admission control; call arrival rate; call blocking; call holding time; call preblocking; cell dwell time; forced termination; quality of service.

1. INTRODUCTION

With increasing demand for mobile computing services and limited available bandwidth, wireless networks increase the number of simultaneous users in the network systems by reducing the cell size. It is projected that future wireless networks will adopt a micro/pico-cellular architecture [1]. However, smaller cell size naturally increases the number of handoffs a MT is expected to make. As the new call arrival rate or *load* increases, so does the probability of handoff failure; this phenomenon, combined with the large number of handoffs before completion of a call, increases the forced termination probability of calls.

Future broadband wireless access systems are expected to integrate various classes of MTs, each class with a different type of quality of service (QoS) requirement. This study considers two classes of MTs—*profiled* MTs [2], and *nonprofiled* or *regular* MTs. The measure of QoS for the profiled MTs is the probability of forced termination of a call that was allowed to access the network. When the load on a wireless network is high, the guarantee of QoS for each class of MTs is a challenging task. Here it is assumed that the trajectory of profiled MTs is known, that is, the sequence of cells a MT crosses during the life of a call is known (say, from a profile database).

Many call-admission control (CAC) algorithms use two generic handoff prioritization schemes: (i) advanced request for a channel—*prerequest scheme* [3], and

---

[1] Department of Computer Science, University of Miami, Coral Gables, Florida 33124.

[2] Department of Computer Science, University of California, Davis, California 95616.

[3] E-mail: sarkar@cs.miami.edu

1068-9605/03/0400-0073/0 © 2003 Plenum Publishing Corporation

(ii) reserving a number of channels for only handoff calls—*guard channel scheme* [3–5]. In general, handoff prioritization schemes decrease handoff failure (and hence forced termination) but increase call blocking. In this work we compare and contrast both the schemes through extensive simulation.

We use two metrics—forced termination probability and successful call completion rate—for assessing the performance of the CAC algorithms. *Forced termination probability (FTP), P_{ft}*, is defined as the ratio of the number of calls forced to terminate because of failed handoff to the number of calls that successfully entered the network. *Successful call completion rate, SCCR*, is defined as the number of calls that are successfully served to completion per unit time by each cell. Note that SCCR is the product of (i) new call arrival rate, (ii) $(1 - P_b)$—probability that a new call is admitted, and (iii) $(1 - P_{ft})$—probability that an ongoing call is not forced to terminate. An ideal CAC algorithm should maintain the value of P_{ft} at or below the desired level, while achieving the highest SCCR. It is found that neither method can guarantee a desired forced termination rate for the profiled MTs. Thus, novel CAC algorithms are necessary for guarantee of QoS to profiled MTs.

In this work, we propose a novel CAC algorithm that can maintain any desired level of QoS while achieving a very high SCCR. In the proposed method, the actual new call arrival rate is estimated continuously, and when the estimated new call arrival rate increases beyond a predetermined level, some new calls are blocked irrespective of the availability of channels. The objective of this preblocking of calls is to maintain the wireless network system's observed new call arrival rate at no more than a predetermined rate.

2. EVALUATION OF HANDOFF PRIORITIZATION SCHEMES

In this section, we evaluate the guard channel and the channel pre-request schemes. We will compare and contrast their merits and demerits.

2.1. Profiled Users

A user profile includes mobility patterns and services accessed. It is recorded against the time of day and is kept in the home location register (HLR) database of the network for all the profiled users [6]. It is assumed that spatial and temporal information (the place and the time of travel) of the profiled users can be obtained from this database. A base station controller (BSC) can use this database to perform profile-based channel allocation decisions, if necessary. The network utilizes profile information in the database for significantly improving the QoS of the profiled MTs by reducing the forced termination probability. Any MT that is not a profiled MT is a *nonprofiled MT*. We assume that the network's knowledge of the profile information is "perfect"; that is, it knows a mobile's trajectory in time and space.

Two generic channel assignment schemes—the guard channel scheme [3–5,9] and the channel prerequest scheme [3]—have been proposed. They can reduce the forced termination probability and improve the QoS. Here, we extensively study them to answer two questions that were not answered in previous studies: (i) How much QoS can they improve for the profiled users *and not all users?* (ii) Can they provide any desired level of QoS to the profiled users under variable loads?

Below, a brief description of each scheme is provided for clarity and completeness.

2.2. Channel Prioritization Schemes

2.2.1. Guard Channel Scheme

In this method, a number of wireless channels, say G out of a total of C channels, called "guard channels," are exclusively reserved for handoff calls of profiled users. The remaining channels, called "normal channels", are shared among all types of calls. By "all types of calls" we mean new calls, the handoff calls of profiled users and the handoff calls of nonprofiled users. New calls and nonprofiled handoff calls are accepted as long as a channel other than the guard channels is available. Profiled handoff calls are accepted until all the channels in the cell are occupied.

2.2.2. Channel Prerequest Scheme

Along the cell boundary, there exists an area where channels of more than one cell can service a MT. The channel prerequest scheme exploits this fact as follows. In the channel pre-request scheme, the neighboring cell to which a profiled user is moving into next can be obtained from the HLR database much before the user leaves the current cell. This information can then be used to pre-request a channel from a neighboring cell a certain amount of time (called the *reservation period*) before leaving the current cell. The reservation period may start at any time if the profiled user is in the region of overlap between cells. It is assumed that trajectories of nonprofiled users are not available and hence channels for them cannot be prerequested. By increasing the reservation period, the probability that a profiled user will be forcefully terminated can be decreased.

2.3. Simulation Model

For our simulation study, we considered hexagonal cells. The cell layout for our simulation is shown in Fig. 1. The 49 white cells are part of the model, and the shaded ones show the wraparound neighbors. The wraparound topology is used because it eliminates the boundary effect keeping exactly six neighbors for each cell [7].

We assume a static channel allocation scheme for cells; that is, the number of channels allocated to a cell does not change during the simulation. All cells are given the same number of channels, and the reuse distance is not a parameter in our study.

2.3.1. Mobility Model

For our work, mobility of MTs is modeled using a simple Brownian-motion or random-walk approximation [10, 11]. In this model, a MT moves to any of the current cell's neighbors with equal probability—1/6 for the hexagonal layout. It is assumed that both profiled and nonprofiled MTs are taking a random walk from cell to cell. However, because the trajectory of each profiled MT is known, the channels for it can be requested before it actually moves to the next cell. Residence time of a MT (profiled or nonprofiled) in a cell is exponentially distributed with mean $1/\eta$ s. All MTs are mobile.

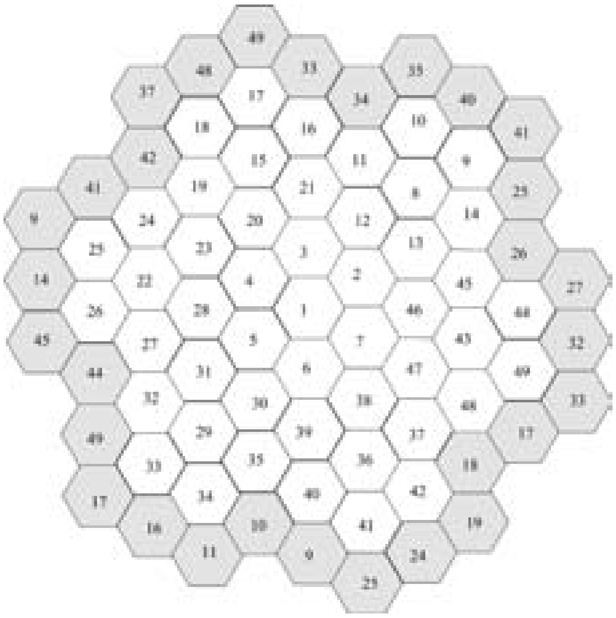


Fig. 1. The cell layout for our simulation.

2.3.2. Additional Modeling Assumptions and Notations

We make the following additional assumptions:

1. We assume a fraction f of the users to be profiled and the remaining $(1 - f)$ to be nonprofiled.
2. New call arrivals into the network follow a Poisson distribution with mean λ calls/s, and the call holding time or the total duration of the call follows an exponential distribution with mean $1/\mu$ s.

Note that these are commonly used assumptions for simulation and analysis. The *load* of a cell is the ratio of call arrival rate to call completion rate, $\rho = \lambda/\mu$ Erlangs/cell.

2.3.3. Default Parameter Values

Unless stated otherwise, we make the following assumptions:

Number of channels/cell	20
Number of calls simulated	300,000
Percentage of profiled users	50
Percentage of nonprofiled users	50
Mean call holding time	120 seconds
Mean call residence time	12 seconds
Reservation period	4 seconds
Number of guard channels/cell	4

2.4. Results and Discussions

To keep labels on the charts short *guard-channels, nonprofiled, profiled, and reservation-period* are denoted by *GC, NP, P*, and *RP*, respectively. Figure 2 shows that the forced termination probability for profiled users and nonprofiled users increases with increasing load in both schemes. This can be attributed to a higher call arrival rate and hence higher channel occupancy.

It is found that SCCR increases in the both channel prerequest and guard channel schemes as the percentage of profiled users changes from 20% to 80% (plots not shown). The reason for this growth can be explained as follows. As the percentage of profiled users increase, the number of users who can get a channel successfully from a neighbor increases.

The probability of forced termination for both profiled and nonprofiled users, however, is found to increase (Fig. 3). The actual growth in P_{ft} is much higher for the nonprofiled users than for the profiled users. For instance,

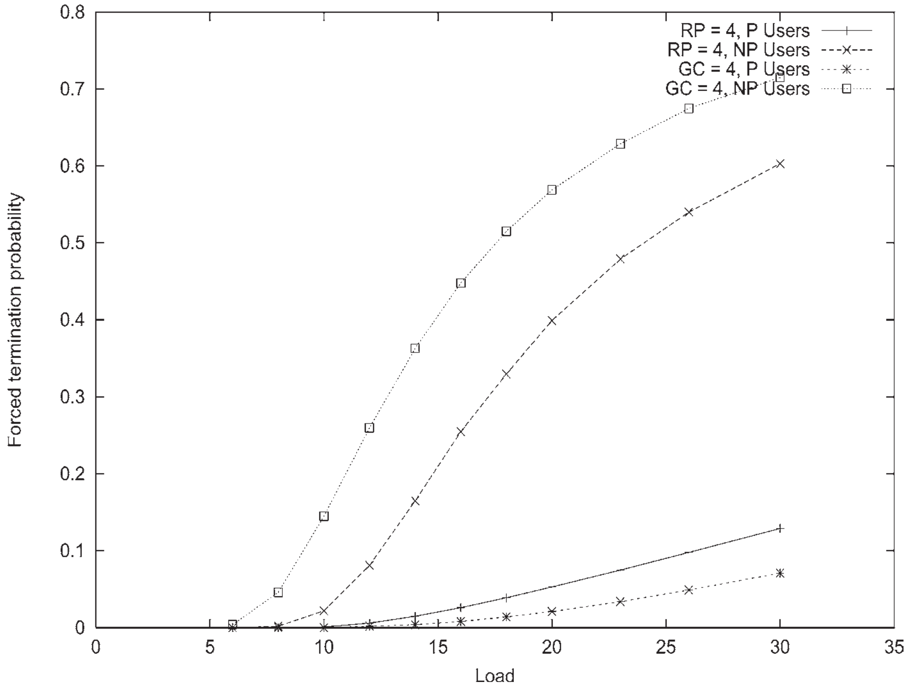


Fig. 2. Variations of P_{ft} between the various types of calls using the two schemes.

at a load of 16 Erlangs with 20% profiled users, P_{ft} for profiled users is 0.0143 and that for nonprofiled users is 0.1972. For the same load, if the percentage of profiled users is increased to 80%, the corresponding forced termination probabilities are 0.04822 and 0.3321. The net effect of this different growth rate is very interesting. Let

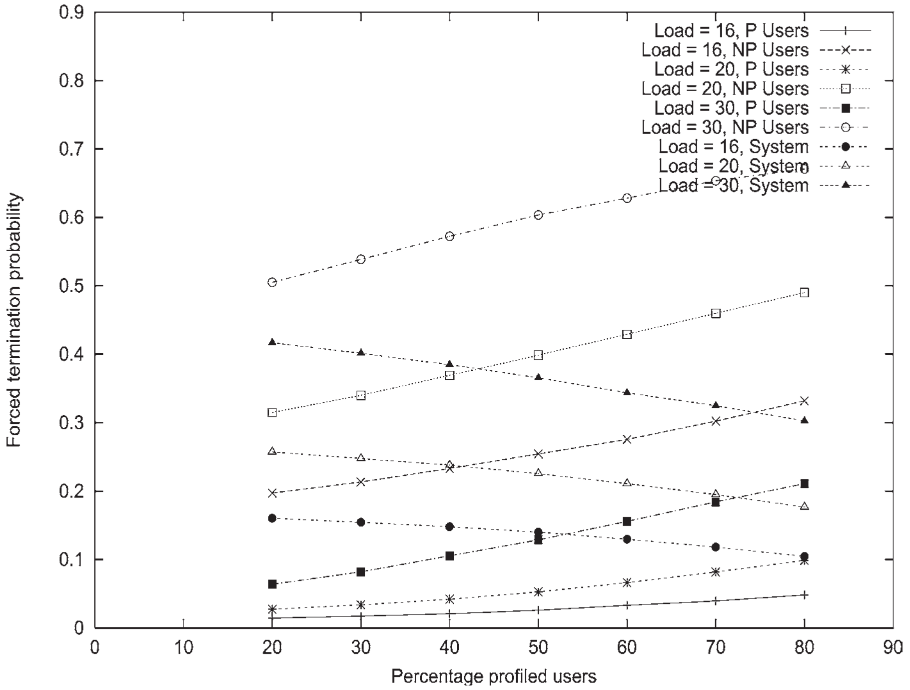


Fig. 3. Comparison of P_{ft} for different types of calls in the channel prerequest scheme.

us compute the weighted average of the forced termination probabilities for the entire system, using the equation,

$$P_{ft}(system) = P_{ft}(profiled) \times f + P_{ft}(nonprofiled) \times (1 - f)$$

The values are 0.1605 and 0.1048 with 20% and 80% profiled users, respectively. We observed that P_{ft} of the system has decreased with an increase in the percentage of profiled users. This naturally increases the SCCR. The successful call completion rate with varying percentage of profiled users in guard channel scheme exhibits the same behavior.

In the channel prerequest scheme, the effect of increasing the reservation period for a constant load is considered next. We observe that, with an increase in reservation time, the P_{ft} of profiled users decreases drastically while that of nonprofiled users increases (Fig. 4). This is due to the fact that, with an increase in the reservation period, the profiled users get a larger time interval for a successful handoff to a neighbor. Hence, their probability of getting a channel from the neighbor increases. This naturally decreases the probability of nonprofiled users getting a channel from the cell. The observations for the guard channel scheme are very similar. In the guard channel scheme, however, it is observed that SCCR decreases continuously with an increase in the number of guard channels (Fig. 5).

From the above discussion, it is evident that the guard channel scheme decreases the forced termination probability of the profiled handoff calls, but it also decreases the successful call completion rate of the system. However, the advantage of this scheme is that it can ensure better QoS to the profiled users if maximum load is known in advance.

The channel prerequest scheme has the advantage of making efficient use of the channels in the system. However, when the load goes beyond a predetermined value, it does not guarantee QoS for the profiled users. For high system utilization, channel prerequest scheme is more desirable.

The schemes discussed above do not always guarantee a desired QoS to profiled users while utilizing the system efficiently. To achieve both, we propose a novel approach that can be used with the guard channel scheme as well as the channel prerequest scheme.

3. THE PROPOSED CALL-ADMISSION CONTROL (CAC) ALGORITHM

Our proposed CAC algorithm controlls the *load observed* by a cell, irrespective of the *actual load* of the system. An acceptable maximum load, ρ_m, of a cell is

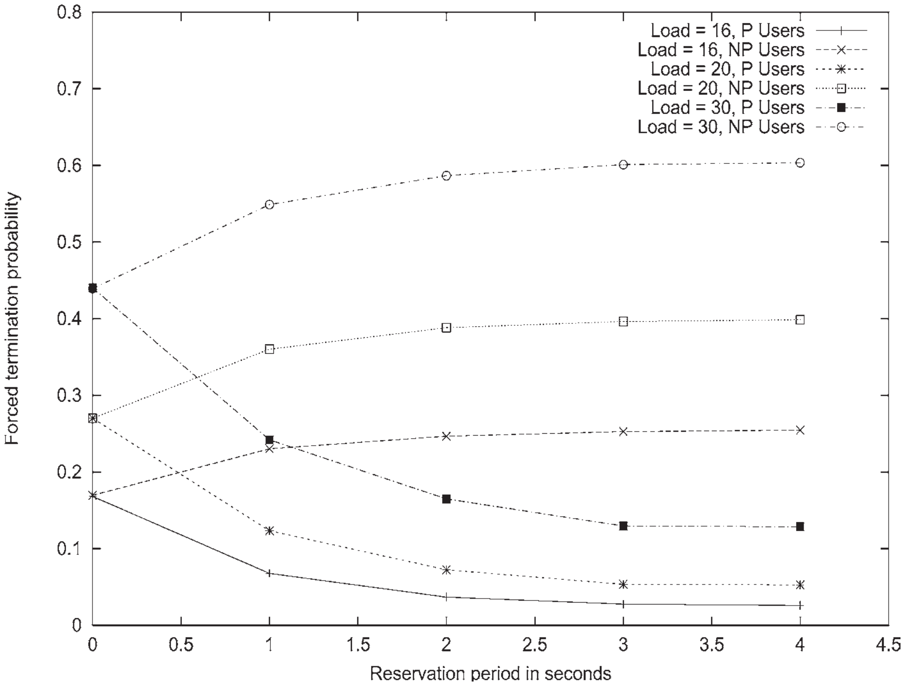


Fig. 4. Impact of reservation period on P_{ft} of calls in the channel prerequest scheme.

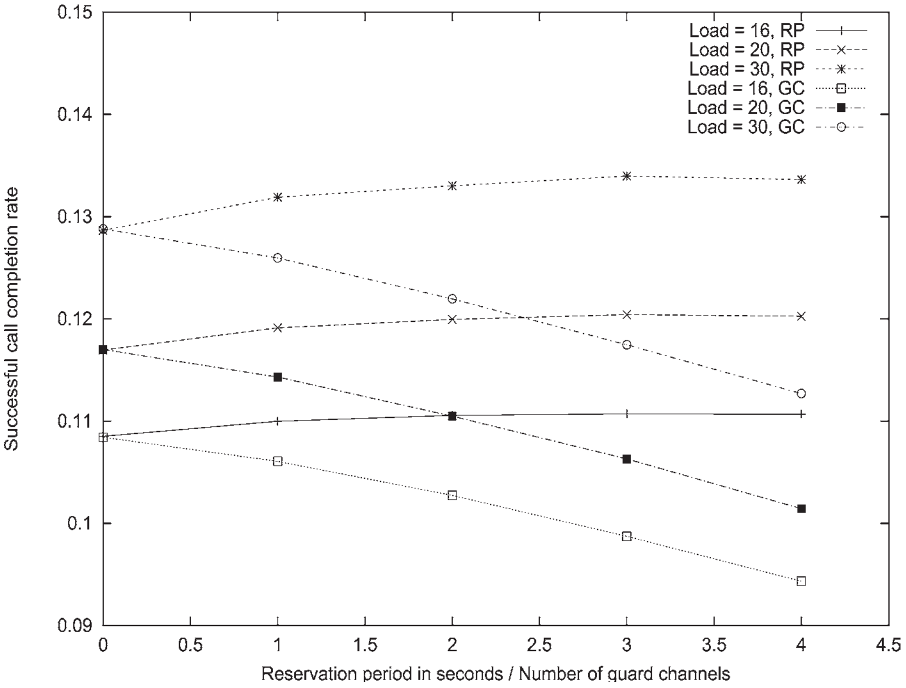


Fig. 5. Changes in SCCR.

determined either by simulations or by an analytical method. The value of ρ_m is used by the CAC algorithm during its operation. Note that, for any load above ρ_m, P_{ft} of the profiled users exceeds the desired value, and hence, the cell fails to meet the desired QoS of the profiled users. A flow chart description of the algorithm is shown in Fig. 6.

During the operation of the system, the arrival rate and hence the expected load is estimated. If the estimated load is no more than ρ_m, attempts are made to allocate channels for all incoming new calls. Otherwise (the load is greater than ρ_m, and) attempts are made to allocate channels for only a fraction, f_r, of the incoming new calls. The other $(1 - f_r)$ fraction of the incoming calls are blocked even when some channels are free. We call this call preblocking. The fraction f_r is calculated as $f_r = \rho_m/\rho_o$, where ρ_o is the estimated load.

There are many possible ways for estimating the load ρ_o. The one used here assumes that information about the arrival times of the most recent N new calls is maintained by the system. Here N is called the "sample size" for estimation of load. If the arrival time of the first new call is t_1 and the arrival time of the last new call is t_N, then $\frac{N-1}{t_N-t_1}$ gives the estimated arrival rate λ. The value of λ/μ, where μ is the service rate of the system, gives the estimated load ρ_o.

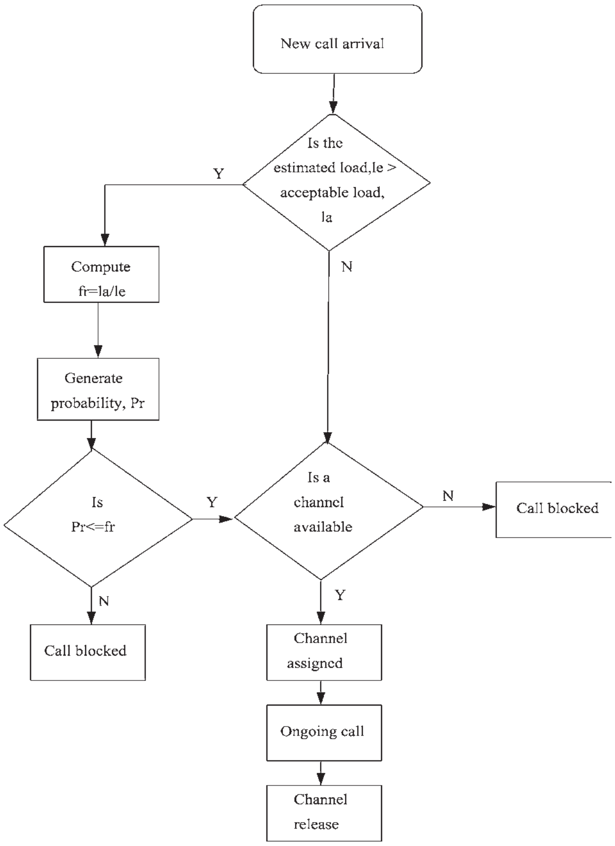


Fig. 6. A flow chart description of the proposed CAC algorithm.

4. PERFORMANCE OF THE PROPOSED CAC ALGORITHM

To conserve space, we report on the performance of the CAC algorithm for a desired value of $P_{ft} = 0.02$ or 2% for the profiled users. From our simulation, it is observed that in the original channel prerequest scheme a load of 14 Erlangs/cell with a reservation period of 4 s satisfies this value of P_{ft}, while in the original guard channel scheme, a load of 14 Erlangs/cell with three guard channels guarantees P_{ft} of profiled users to be below 2%. So we assume that a load of 14 Erlangs/cell is the acceptable maximum load in the proposed method. Default sample size for arrival rate estimation is 50. We have tested the CAC algorithm for many other combinations of parameter values and have observed similar performance.

Figure 7 shows the variation of the forced termination probability with increasing load, when the CAC algorithm is used with the prerequest scheme. The forced termination probability increases for profiled as well as for nonprofiled users up to a certain load and then becomes almost constant. At low load, the system accepts almost all new calls. However, the network blocks some new calls if the estimated load is greater than 14 Erlangs/cell. This call preblocking always maintains the observed load in a cell to be about 14 Erlangs/cell. This stabilizes the forced termination probability of all types of users at a load greater than 14 Erlangs/cell. Observe that the P_{ft} for profiled users remains at or below 2%, even though the load increases. We tested even with such (unreasonably) high load as 300 Erlangs/cell, and the performance was as good!

In Fig. 8, the variation of SCCR with load is shown when the CAC algorithm is incorporated in the channel prerequest scheme. Successful call completion rate initially increases with load, up to a load 14 of Erlangs/cell, and then becomes almost constant. When the estimated load is below 14 Erlangs/cell, the successful call completion rate increases with load, because no call or very few calls are rejected without any attempt to allocate channel. The reason for SCCR becoming almost constant above a load of 14 Erlangs/cell is the following. To keep the P_{ft} of profiled users below 2%, new calls are not admitted into the system when the estimated load goes above 14 Erlangs/cell and so P_b increases and $(1 - P_b)$ decreases. But, because the time taken for new call arrivals at high load is lower than the time taken for new call arrivals at low load for the same number of calls, the number of calls successfully served per unit time is almost constant for any offered load above 14 Erlangs/cell.

When we tried to look at the effect of incorporating the CAC algorithm in the guard channel scheme, we

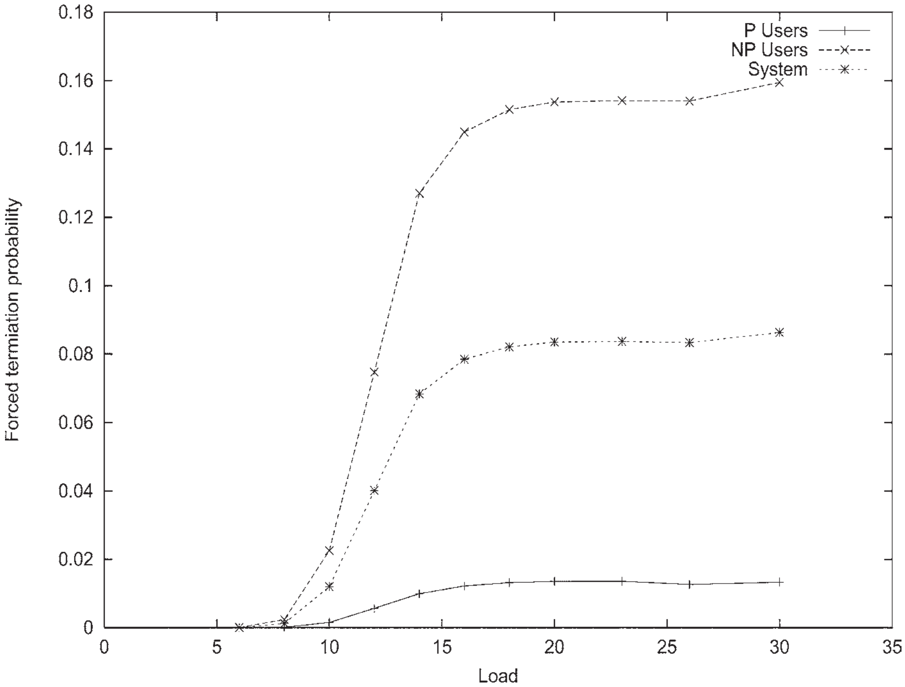


Fig. 7. P_{ft} in the system when the CAC algorithm is incorporated into the channel prerequest scheme.

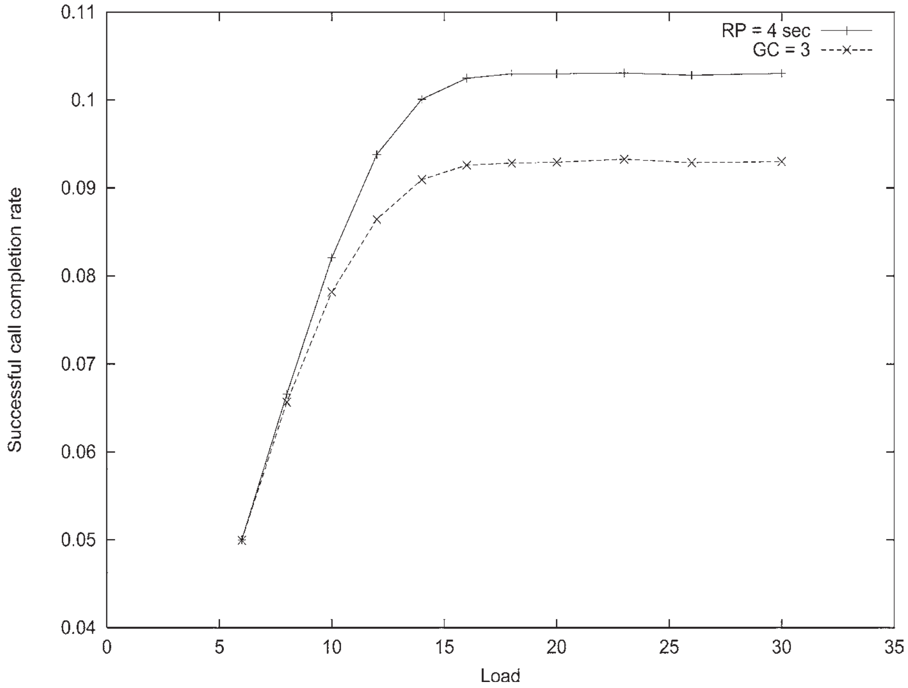


Fig. 8. Difference in SCCR when the CAC algorithm is incorporated into the original schemes.

observed that P_{ft} and SCCR exhibited similar behavior. However, SCCR in guard channel scheme is lower than that of channel prerequest scheme when the CAC algorithm is used. This behavior is depicted in Fig. 8. This can be attributed to the inefficient utilization of guard channels in the guard channel scheme.

Figure 9 shows the percentage fall in the successful call completion rate with load, when CAC algorithm is incorporated in the channel prerequest scheme, compared to SCCR in a system without CAC algorithm at a load of 14 Erlangs/cell. We see that the difference is higher at a load of 14 Erlangs/cell and then tends to decrease and almost stabilizes after a load of 18 Erlangs/cell. This behavior is naturally expected from the behavior of SCCR with load.

When the CAC algorithm is incorporated in the channel prerequest scheme, the blocking probability of the system increases faster with load. Over a load of 14 Erlangs/cell, we observe an increase in the blocking probability of the system using the proposed method, when compared to the blocking probability of the system using the original channel prerequest scheme. This is expected because of the rejection of a fraction of new calls, even if a channel is available, when the load is above 14 Erlangs/cell.

From the above discussion, it is evident that the CAC algorithm can guarantee QoS to the profiled users. Moreover, it maintains the successful call completion rate of the system to be close to the maximum value. We now discuss the effect of sample size (for estimation of new call arrival rate) on the system parameters.

5. THE EFFECT OF SAMPLE SIZE

The simulation study reported in the previous section used arrival times of the most recent 50 calls for estimating call arrival rates. In this section we study the effect of *sample size* on the estimation of arrival rates. We use sample size to mean the number of call arrival times used for the estimation of arrival rates. Figure 10 shows plots of call blocking probabilities with load for sample sizes 20, 30, 40, and 50. It is observed that for the same load, an increase in sample size causes a decrease in new call blocking probability. Plots in Fig. 11 show call force termination probability for profiled uses and nonprofiled users and the system for sample sizes 20 and 50. Again, a clear and consistent decrease in forced termination probability is observed for a decrease in sample size. (We have shown plots for only two sample sizes to keep figures legible.) Plots in Fig. 12 show SCCR for sample sizes 20, 30, 40, and 50. For every load we see a consistent pattern—the smaller the sample size used, the *better* the QoS (or *smaller* the probability of forced termination)

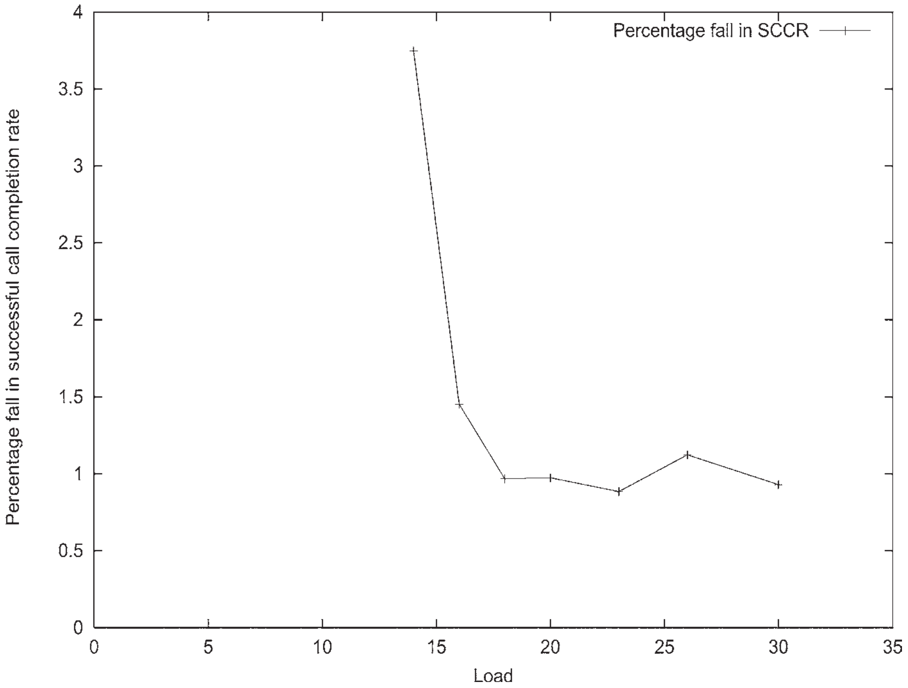


Fig. 9. Percentage fall in SCCR when the CAC algorithm is incorporated into the channel prerequest scheme, compared to SCCR without the CAC algorithm at load 14.

for both the classes of calls. The cost for a smaller sample size (or better QoS) is a *higher* call blocking probability (see Fig. 10), and hence a *lower* SCCR for the overall system (see Fig. 12). In this section we show that this is not an artifact of our simulation, but an inherent nature of the Poisson arrival process.

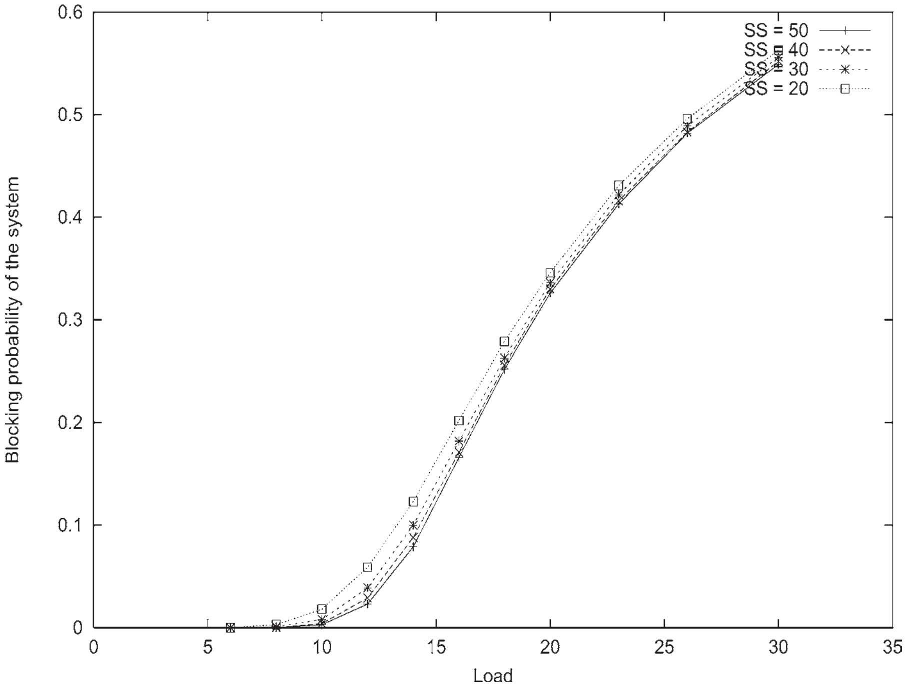


Fig. 10. Variation of P_b with sample size.

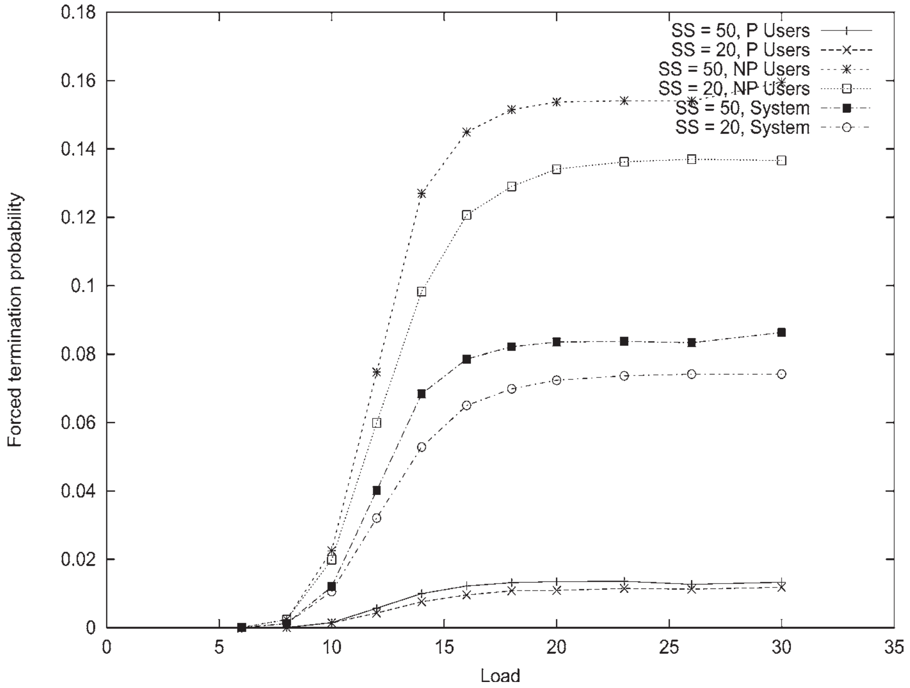


Fig. 11. Decrease in the P_{ft} with decrease in the sample size.

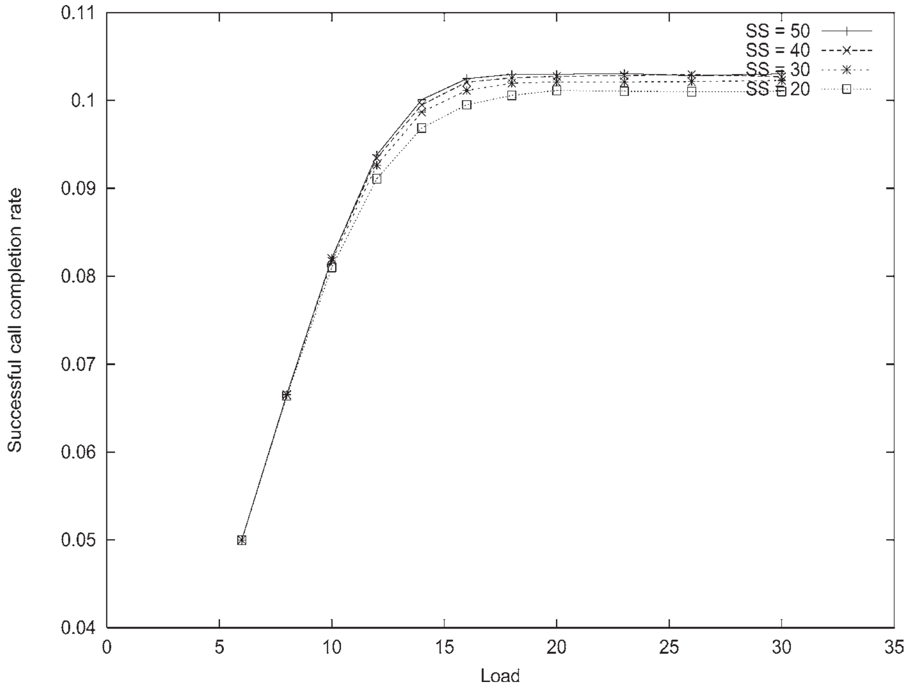


Fig. 12. Variation of SCCR of the overall system with sample size.

5.1. Estimation of Arrival Rates with Finite Number of Samples

If arrivals follow a Poisson distribution with λ call/s, the probability of exactly k arrivals during a time interval of length t can be written as [12]

$$p(k,\lambda,t) = \frac{e^{-\lambda t}(\lambda t)^k}{k!} \tag{1}$$

Now consider a time interval of length $\frac{1}{\lambda} \times n = \frac{n}{\lambda}$; that is, product of an average time between two consecutive calls and n. The probability of arrival of k calls during this length of time is obtained by substituting $\frac{n}{\lambda}$ for t in Eq. (1).

$$p\left(k,\lambda,\frac{n}{\lambda}\right) = \frac{e^{-\lambda\frac{n}{\lambda}}\left(\lambda\frac{n}{\lambda}\right)^k}{k!} \tag{2}$$

After simplification, we obtain

$$p\left(k,\lambda,\frac{n}{\lambda}\right) = \frac{e^{-n}n^k}{k!} \tag{3}$$

An interesting fact is that the right-hand side of Eq. (3) is independent of call arrival rate λ. Thus the discussions that follow are independent of call arrival rate, and hence we use $p(k,n)$ for $p(k,\lambda,\frac{n}{\lambda})$.

The probability of arrival of fewer that n calls during any interval of time length $\frac{n}{\lambda}$ is obtained by summing the right-hand side of Eq. (3).

$$p(n) = \sum_{k=0}^{n=1}\frac{e^{-n}n^k}{k!} = e^{-n}\sum_{k=0}^{n-1}\frac{n^k}{k!} \tag{4}$$

The sum on the right-hand side of Eq. (4) is the probability of *underestimation* of arrival rates, because this is the probability that fewer than n calls arrive during a time duration $\frac{n}{\lambda}$.

A plot of $p(n)$ with sample size is shown in Fig. 13. It is clear that $p(n)$ is smaller than 0.5 for finite values of n; $p(n)$ gradually increases and approaches 0.5 as n goes to ∞, as the *law of large numbers* dictates. Thus, the smaller the sample size, the lower is the probability of underestimation of the call arrival rate. In other words, the smaller the sample size, the higher is the probability of overestimation of arrival rate. With smaller sample size, the consistent improvement in QoS for both classes of calls (observed in Fig. 11) is a natural phenomenon of the Poisson arrival process. Furthermore, use of any sample size is bound to keep the QoS at a predefined level, when it is evaluated over a long period.

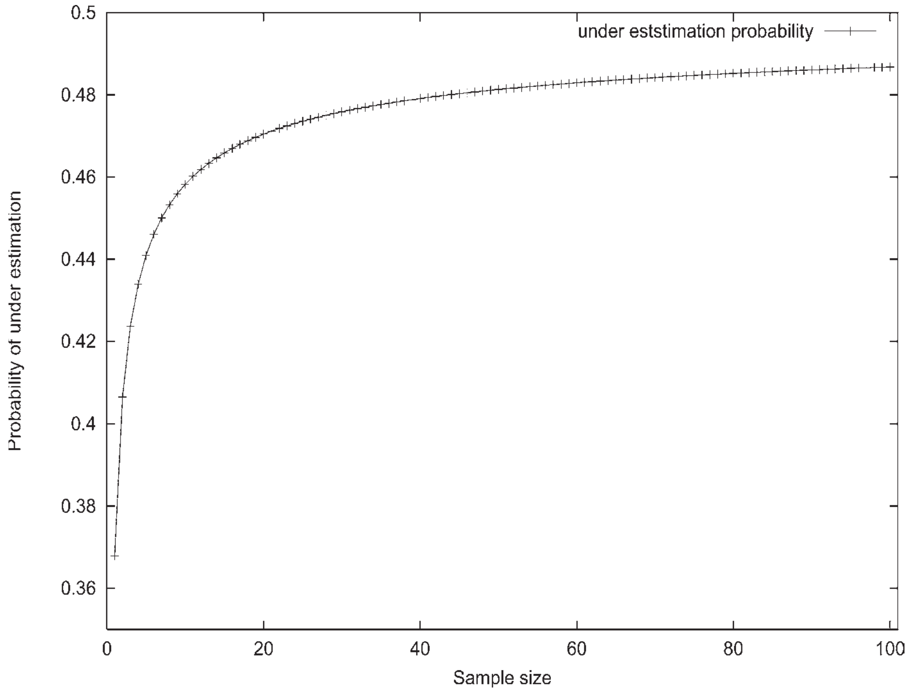


Fig. 13. Sample size and probability of under estimation of call arrival rate.

6. CONCLUSION

In this paper, we evaluated two handoff prioritization methods for channel allocation. We compared and contrasted their advantages and disadvantages. We found that they fail to meet the desired QoS when the system load exceeds a predetermined value. We have proposed a novel CAC algorithm that guarantees QoS, that is, forced termination probability for profiled users at any load. From the network operator's view, the algorithm can maintain a very high successful call completion rate. The proposed CAC algorithm can work independently or in conjunction with other handoff prioritization schemes.

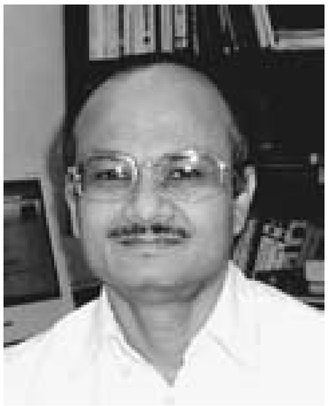
The system using the CAC algorithm works just like the system using the original scheme. The only differences are that (i) when the estimated load ρ_o is below a certain value ρ_m, the proposed system accepts all new calls, and (ii) when the estimated load ρ_o goes above the value ρ_m, it accepts the new calls selectively.

REFERENCES

1. A. S. Acampora and M. Naghshineh, "Control and Quality of Service Provisioning in High Speed Microcellular Networks," *IEEE Pers. Commun.*, Vol. 1, No. 2, pp. 36–43, 1994.
2. A. N. Rudrapatna and C. Giardina, "Channel Occupancy and Network Utilization Implications for a Profile-Driven Resource Allocation Scheme in Cellular Networks," in *Proceedings of Globecom*, pp. 2000–2011, November 1998.
3. Y. B. Lin, S. Mohan, and A. Noerpel, "Queueing Priority Channel Assignment Strategies for PCS Hand-off and Initial Access," *IEEE Trans. Veh. Technol.*, Vol. 43, pp. 704–712, No. 3, August 1994.
4. D. Hong and S. S. Rappaport, "Traffic Model and Performance Analysis for Cellular Mobile Radio Telephone Systems with Prioritized and Non-prioritized Handoff Procedures," *IEEE Trans. Veh. Technol.*, Vol. VT-35, No. 3, pp. 77–92, August 1986.
5. Y. C. Kim, D. E. Lee, B. J. Lee, Y. S. Kim, and B. Mukherjee, "Dynamic Channel Reservation Based on Mobility in Wireless ATM Network," *IEEE Commun. Mag.*, pp. 47–51, Nov. 1999.
6. V. Pandey, D. Ghosal, and B. Mukherjee, "Quantifying the Benefits of Exploiting User Profiles in Cellular Networks," in *Proc. ICPWC'99*, India, pp. 374–378, February 1999.
7. L. O. Guerrero and A. H. Aghvami, "A Prioritized Handoff Dynamic Channel Allocation Strategy for PCS," *IEEE Trans. Veh. Technol.*, Vol. 48, No. 4, pp. 1203–1215, July 1999.
8. O. T. W. Yu and V. C. M. Leung, "Adaptive Resource Allocation for Prioritized Call Admission over an ATM-Based Wireless PCN," *IEEE J. Select. Areas Commun.*, Vol. 15, No. 7, pp. 1208–1225, September 1997.
9. R. Ramjee, D. Towsley, and R. Nagarajan, "On Optimal Call Admission Control in Cellular Networks," *ACM/Baltzer Wireless Netw. (WINET)*, Vol. 3, pp. 29–41, March 1997.
10. Y. B. Lin, L. F. Chang, and A. Noerpel, "Modeling Hierarchical Microcell/Macrocell PCS Architecture," *Proc. ICC*, (Seattle, WA), pp. 405–409, 1995.
11. A. D. May, *Traffic Flow Fundamentals*, Prentice Hall, 1990.
12. S. Ross, *A First Course in Probability*, Prentice Hall, 1998.

Biswanath Mukherjee received the B.Tech. (Hons) degree from Indian Institute of Technology, Kharagpur (India) in 1980 and the Ph.D. degree from University of Washington, Seattle, in June 1987. At Washington, he held a GTE Teaching Fellowship and a General Electric Foundation Fellowship. In July 1987, he joined the University of California, Davis, where he has been Professor of Computer Science since July 1995 and served as Chairman of Computer Science from September 1997 to June 2000. He is co-winner of paper awards presented at the 1991 and the 1994 National Computer Security Conferences. He serves or has served on the editorial boards of the *IEEE/ACM Transactions on Networking, IEEE Network, ACM/Baltzer Wireless Information Networks (WINET), Journal of High-Speed Networks, Photonic Network Communications, and Optical Network Magazine*. He also served as Editor-at-Large for optical networking and communications for the IEEE Communications Society. He served as the Technical Program Chair of the IEEE INFOCOM '96 conference. He is author of the textbook *Optical Communication Networks* published by McGraw-Hill in 1997, a book that received the Association of American Publishers, Inc.'s 1997 Honorable Mention in Computer Science. He is a Member of the Board of Directors of IPLocks, Inc., a Silicon Valley startup company. He has consulted for and served on the technical advisory boards of a number of startup companies in optical networking. His research interests include lightwave networks, network security, and wireless networks.

Satya Kovvuri got her Bachelors degree in Computer Engineering from GITAM., Andhra University (India) in 1998. She got her M.S. degree in computer science from University of Miami and is currently working as a financial software developer in Bloomberg LP in New York.



Dilip Sarkar received the B.Tech. (Hons.) degree in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, India, in May 1983, the M.S. degree in computer science from the Indian Institute of Science, Bangalore, India, in December 1984, and the Ph.D. degree in computer science from the University of Central Florida, Orlando, in May 1988. From January 1985 to August 1986 he was a Ph.D. student at Washington State University, Pullman. He is currently an Associate Professor of Computer Science at the University of Miami, Coral Gables. His research interest include design and analysis of algorithms, parallel and distributed processing, middleware and web computing, multimedia communication over broadband and wireless networks, fuzzy systems, and neural networks. In these areas, he has guided several theses and has authored numerous papers. Dr. Sarkar is a senior member of the IEEE, a member of IEEE Communications Society and the Association for Computing Machinery. He was a recipient of the Fourteenth All India Design Competition Award in electronics in 1982.

Vijoy Pandey (vijoy@nortelnetworks.com) received a B.Tech. (Hons.) degree from Indian Institute of Technology, Kharagpur, in 1995, and an M.S. degree from University of California, Davis in 1997. He is currently pursuing his Ph.D. degree at UC Davis, while working in the Ethernet Switching Group at Nortel Networks in Santa Clara, California. At UC Davis, he was nominated for the Professors for the Future Fellowship Award in 1999. His research interests include architectures and protocols for next-generation wireless, cellular networks, and next-generation Ethernet switching.

Dipak Ghosal received his B.Tech degree in Electrical Engineering from Indian Institute of Technology, Kanpur, India, in 1983, MS degree in Computer Science from Indian Institute of Science, Bangalore, India, in 1985, and Ph.D. degree in Computer Science from University of Louisiana, Lafayette, USA, in 1988. From 1988 to 1990 he was a Research Associate at the Institute for Advanced Computer Studies at University of Maryland (UMIACS) at College Park, USA. From 1990 to 1996 he was a Member of Technical Staff at Bell Communications Research (Bellcore) at Red Bank, USA. Currently, he is with the faculty of the Computer Science Department at the University of California at Davis, USA. His research interests are in the areas of peer-to-peer systems, mobile, adhoc, and cellular networks, IP telephony, and performance evaluation of computer and communication systems.