Joint Connection-Level and Packet-Level Quality-of-Service Support for VBR Traffic in Wireless Multimedia Networks

Lei Huang, *Member, IEEE,* and C.-C. Jay Kuo, *Fellow, IEEE*

***Abstract*—In this paper, we investigate call admission control (CAC) schemes that can jointly provide connection-level quality-of-service (QoS) (in terms of the new call blocking probability and the handoff dropping probability) and packet-level QoS (in terms of the packet loss probability) for wireless multimedia networks. Stationary CAC schemes are proposed as the results of the solution to constrained optimization problems. A dynamic CAC scheme that can be adapted to varied and varying traffic conditions dynamically is also proposed. The proposed CAC schemes are computationally efficient and easy to implement, thus being suitable for real-time system deployment. Simulation results have demonstrated that the proposed dynamic CAC scheme achieves better performance when applied to realistic traffic conditions found in wireless multimedia networks.**

***Index Terms*—Call admission control (CAC), handoff dropping probability, new call blocking probability, packet loss probability, quality-of-service (QoS), wireless multimedia network.**

I. Introduction

QUALITY-OF-SERVICE (QoS) provisioning in wireless networks presents great challenge due to the limited bandwidth resource, the highly variable environment and user's mobility. Traditional wireless communication networks use the circuit-switching technology for voice communications, where the primary concern of QoS is the service connectivity and continuity at the connection-level. The connection-level QoS is usually measured by the new call blocking probability and the handoff dropping probability. With the explosive increase of the wireless communication demand, the next-generation wireless networks will employ the packet-switching technology to support multimedia services. Most multimedia applications are with variable-bit rate (VBR) traffic, which allows better utilization of radio resources and higher efficiency of the total system via multiplexing among services. However, packets of certain services may experience varying delay, delay jitter and loss, which will have a great impact on the application quality perceived by end users. Therefore, the packet-level QoS, in terms of delay/jitter and packet loss probability, has to be guaranteed as well. In this paper, we investigate the QoS provisioning problem at both the connection-level and the packet-level for multimedia applications in the next-generation wireless networks.

Call admission control (CAC) schemes have been investigated extensively in traditional circuit-switching wireless networks as an important mechanism to support connection-level QoS without considering packet-level QoS [1]–[6]. On the other hand, most of the previous research efforts on packet-level QoS [7], [8] did not address the handoff and mobility management issues that are important for connection-level QoS provisioning.

More recently, there has been research work addressing both connection-level and packet-level QoS for wireless networks. The resource reservation and call admission control scheme proposed in [9] considered delay by differentiating real-time and nonreal-time traffic, but did not provide quantitative packet-level QoS guarantee analysis. Meanwhile, it reserved resources for real-time traffic in all neighboring cells to provide connection-level QoS guarantee for real-time traffic, which is not efficient as the mobile user only hands off to one of these neighbors. In [10], an adaptive QoS handoff priority scheme was analyzed in both connection-level QoS (i.e., the new call blocking probability and the handoff dropping probability) and packet-level QoS (i.e., delay) performance. Since it deals with these two level QoS problems separately, there is no guarantee or optimization on the overall QoS performance. Talukdar *et al.* [11] proposed an admission control scheme based on a new three-class service model for integrated services packet networks with mobile hosts. It extended the InteServ architecture [12] for QoS provisioning in wired Internet protocol (IP) networks to mobile hosts. This scheme aimed at a guaranteed packet scheduling delay bound at the packet-level, but had limitations in connection-level QoS provisioning since it required that each mobile host requesting a connection should provide its accurate mobility specification, which consists of the set of cells the mobile host is expected to visit during the lifetime of the connection. In all schemes mentioned above, only delay was considered as the packet-level QoS metric, while the packet loss probability was not addressed. Several joint admission and congestion control schemes have been proposed for integrated voice and data services for packet radio networks [13], [14]. The numbers of voice calls and data message connections to be admitted to the system were decided according to the call blocking probability of voice calls and the delay of data packets, respectively. However, priority of handoff calls and data packet loss are not considered. For code-division multiple-access (CDMA) wireless networks with interference-limited soft capacity, call admission control

Manuscript received January 29, 2004, revised December 8, 2004.

L. Huang is with the Department of Electrical Engineering and Computer Science, Loyola Marymount University, Los Angeles, CA 90045 USA (e-mail: lhuang@lmu.edu).

C.-C. J. Kuo is with the Integrated Media Systems Center and Department of Electrical Engineering-Systems, University of Southern California, Los Angeles, CA 90089-2564 USA (e-mail: cckuo@sipi.usc.edu).

Digital Object Identifier 10.1109/JSAC.2005.845626

strategies based on the signal-to-interference ratio (SIR) measurement were proposed to support the single service [15], as well as multimedia services [16]. These schemes considered both physical-layer SIR constraints and connection-level QoS simultaneously. However, they did not consider the packet-level QoS.

The wireless networks of our interest in this work is the last-hop type, including both wireless LANs and WANs, where the infrastructure consists of a wireless access network and a wired backbone network such as the Internet. Although the wireless access network may result in longer delay in general, the major part of delay jitter comes from the multihop wired Internet. QoS in this aspect has been investigated extensively in the context of IP networks. Extension to the last hop of wireless access has also been studied [11]. For this reason, we consider the packet loss probability as the primary packet-level QoS metric. Generally speaking, the smaller the packet loss is, the higher quality could be perceived by the end user. In this context, packet-level and connection-level QoS becomes a trade-off, and their joint optimization is important for the overall QoS provisioning.

In this paper, we propose efficient CAC schemes that can jointly provide QoS at both connection and packet levels for emerging packet-switching wireless networks under the constraint of limited and varying bandwidth resources. The major contributions of this work are highlighted below.

- Both connection-level QoS in terms of the new call blocking probability and the handoff dropping probability and packet-level QoS in terms of the packet loss probability are analyzed based on a common traffic model. It provides insights into the relationship and tradeoff between QoS provisioning at these two levels.
- Resource allocation is formulated as a joint optimization problem with different objectives and constraints on QoS provisioning.
- Suboptimal solutions to the formulated problems are developed, and the corresponding stationary CAC schemes are proposed to achieve the objectives.
- A dynamic CAC scheme that is adaptive to varied and varying traffic conditions is proposed.
- All the proposed CAC schemes are computationally efficient and easy to implement, thus being suitable for real-time system deployment.

The rest of the paper is organized as follows. In Section II, stationary CAC schemes with joint connection-level and packet-level QoS optimization are investigated based on a traffic model. In Section III, a dynamic CAC scheme is proposed to take the traffic dynamics of real-world networks into account. Experimental results are given in Section IV, where the performances of the proposed schemes are evaluated and compared with simulations. Finally, Section V concludes the paper and presents some future work directions.

II. Stationary CAC Schemes for VBR Traffic

In a wireless network, packet loss occurs at different layers. At the physical layer, the error-prone radio channel causes data corruption. Advanced channel coding schemes have been developed to solve this problem [17]–[19]. At the link layer, competing access to the shared radio resource results in data collision. Multiple access control (MAC) schemes with QoS considerations have been developed to avoid this problem [20], [21]. At the network layer, packet loss is mainly caused by congestions due to the scarce of network resources. Our work addresses this problem and develops the network layer CAC mechanism to jointly optimize packet-level and connection-level QoS. In this section, we consider stationary CAC policies, i.e., they depend only on the current system state, which is defined by the number of VBR connections (including new and handoff) being carried in the cell. Without loss of generality, we examine one single cell with the assumption that all other cells have a similar behavior statistically.

A. VBR Traffic Model

For each VBR traffic flow, a two-state on–off model is used to describe the source traffic. It is assumed in this model that the source traffic is in either the on- or the off-state. The data traffic is generated at the peak rate in the on-state, while no traffic is generated in the off-state. The probability of the source in the on-state is denoted by $P_{\rm on}$. Furthermore, we assume that all VBR connections have the unit peak rate, and are independent of each other. Mathematically, the ith VBR traffic is a random process $X_i(t)$, which consists of independent identically distributed random variables for any i and t with $P\{X_i(t)=1\} = P_{\rm on}$ and $P\{X_i(t)=0\} = 1-P_{\rm on} = P_{\rm off}$. In the following, we use this VBR traffic model to analyze the packet loss probability and its properties, which provides insights into the development of our call admission control schemes. The simple model is chosen for its mathematical tractability with closed-form solutions. However, our call admission schemes can be extended to more realistic multimedia VBR traffic, provided that the packet loss probability follows the same properties.

The current active load $L(t)$, defined as the total traffic in the on-state, is a random process obtained by summation of all the carrying connections. That is

$$L(t) = \sum_{i=1}^{n} X_i(t) \tag{1}$$

where n is the total number of VBR connections being carried in the system. For any t, $L(t)$ has a binomial distribution $B(n, P_{\rm on})$, i.e.,

$$P\{L(t)=k\} = \binom{n}{k} P_{\rm on}^k (1-P_{\rm on})^{n-k}. \tag{2}$$

Suppose that the system capacity is C. Then, the average load loss is the expectation of $L(t)$ exceeding C, which can be written as

$$E(L(t)-C) = \sum_{m=1}^{n-C} mP\{L(t)=C+m\}. \tag{3}$$

The average total active load is

$$E(L(t)) = \sum_{k=1}^{n} kP\{L(t)=k\} = nP_{\rm on}. \tag{4}$$

Thus, the load loss rate $R_L(n)$ with n calls in the system becomes

$$R_L(n) = \frac{E(L(t) - C)}{nP_{\text{on}}}. \tag{5}$$

Assume that the new and the handoff connection arrivals are Poisson distributed with rates λ_n and λ_h, respectively. The cell residence time of each connection is exponentially distributed with mean $1/\eta$. Let $\lambda = \lambda_n + \lambda_h$, and $\lambda_n = \alpha\lambda$. The corresponding Markov chain gives the stationary state distribution as

$$\pi(n) \propto \frac{\left(\frac{\lambda}{\eta}\right)^n}{n!} \prod_{i=1}^{n} (\alpha P_{an}(i-1) + (1-\alpha) P_{ah}(i-1)) \tag{6}$$

such that

$$\sum_{i=0}^{n} \pi(i) = 1 \tag{7}$$

where $P_{an}(i)$ and $P_{ah}(i)$, $i = 0, 1, \ldots, n$, denote the acceptance probabilities for the new call and the handoff call requests, respectively, when the system is in state i.

Consequently, the QoS metrics P_B, P_D, and P_L are all functions of the CAC policy ϕ, defined by $P_{an}(i)$ and $P_{ah}(i)$, $i = 0, 1, \ldots, N$, where N is the maximum number of connections allowed in the cell. The new call blocking probability is given by

$$P_B = \sum_{i=0}^{N} \pi(i)(1 - P_{an}(i)) \tag{8}$$

the handoff dropping probability is

$$P_D = \sum_{i=0}^{N} \pi(i)(1 - P_{ah}(i)) \tag{9}$$

and the packet loss probability caused by traffic congestion is

$$P_L = E(R_L(N)) = \sum_{i=0}^{N} \pi(i) R_L(i). \tag{10}$$

B. Stationary CAC Problem Formulation

The stationary CAC problem is to decide the unknown variables $P_{an}(i)$ and $P_{ah}(i)$, such that P_L, P_D, and P_B are optimized jointly. In particular, the following three problems are formulated with different optimization objectives.

1) Minimize P_L subject to P_B and P_D constraints. This problem is to obtain the best media application quality by minimizing the packet loss probability under certain constraints on the connection-level QoS. Mathematically, we would like to find the optimal CAC policy ϕ^* such that $P_L(\phi^*) = \min P_L(\phi)$, subject to $P_B \le P_B^{\text{tar}}$, and $P_D \le P_D^{\text{tar}}$.
2) Minimize P_B subject to P_D and P_L constraints. This problem is to find the optimal CAC policy ϕ^* such that $P_B(\phi^*) = \min P_B(\phi)$, subject to $P_D \le P_D^{\text{tar}}$, and $P_L \le P_L^{\text{tar}}$.
3) Minimize P_D subject to P_B and P_L constraints. This problem is to find the optimal CAC policy ϕ^* such that $P_D(\phi^*) = \min P_D(\phi)$, subject to $P_B \le P_B^{\text{tar}}$, and $P_L \le P_L^{\text{tar}}$.

Since the state space could be infinite as N increases indefinitely, it is not practical for a real-time system to solve the above constrained optimization problems for optimal solutions. Instead, suboptimal solutions are found within a subset of the stationary CAC schemes to reduce the computational complexity dramatically.

C. Suboptimal Solutions

We consider the set of CAC schemes $\{\phi(N,T)\}$, where T is a real number, N is an integer, and $T \le N$. The acceptance probabilities of new call requests (P_{an}) are

$$P_{an}(i) = \begin{cases} 1, & 0 \le i < \lfloor T \rfloor \\ 0, & \lfloor T \rfloor + 1 \le i \le N \\ T - \lfloor T \rfloor, & i = \lfloor T \rfloor \end{cases} \tag{11}$$

where $\lfloor T \rfloor$ is the largest integer number not exceeding T. The acceptance probabilities of handoff requests (P_{ah}) are

$$P_{ah}(i) = \begin{cases} 1, & 0 \le i < N \\ 0, & i = N. \end{cases} \tag{12}$$

In other words, the CAC policy admits a new connection request if and only if the current load is less than T, while admits a handoff request if and only if the current load is less than N. When T is a noninteger, the CAC policy admits a new connection request with the probability of the fractional part of T. This subset of CAC is called the limited fractional guard channel scheme, which has been proved to be the optimal CAC policy for constrained connection-level QoS provisioning [1]. Within this subset, the optimization of CAC schemes is to find the optimal values of N and T such that the objectives are satisfied.

Let β denote $T - \lfloor T \rfloor$, and $\rho = \lambda/\eta$. For a given set of (N,T), the steady-state distribution is shown in (13) at the bottom of the page, where

$$P_0(N,T) = \left(\sum_{i=0}^{\lfloor T \rfloor} \frac{\rho^i}{i!} + \sum_{i=\lfloor T \rfloor + 1}^{N} \frac{\rho^i (\alpha\beta + (1-\alpha))(1-\alpha)^{i-\lfloor T \rfloor - 1}}{i!} \right)^{-1}. \tag{14}$$

$$\pi_{N,T}(i) = \begin{cases} \frac{\rho^i}{i!} P_0(N,T), & 0 \le i \le \lfloor T \rfloor \\ \frac{\rho^i (\alpha\beta + (1-\alpha))(1-\alpha)^{i-\lfloor T \rfloor - 1}}{i!} P_0(N,T), & \lfloor T \rfloor + 1 \le i \le N \end{cases} \tag{13}$$

Now, we can rewrite the QoS metrics P_B, P_D and P_L as functions of (N,T), as shown in (15) and (16) at the bottom of the page, and

$$P_L(N,T)=\sum_{i=0}^{N}\pi_{N,T}(i)R_L(i) \tag{17}$$

where $R_L(i)$ is shown in (18) at the bottom of the page.

From the previous equations, we can derive the following properties of the three QoS metrics P_B, P_D, and P_L.

- Property 1: For a given N, P_B decreases with the increase of T.
  If $T1>T2$, then $P_B(N,T1)<P_B(N,T2), \forall\, N\geq T1$.
- Property 2: For a given T, P_B increases with the increase of N and approaches to upper limit P_B^U.
  If $N1>N2$, then $P_B(N1,T)>P_B(N2,T), \forall\, T\leq N2$; and $\lim_{N\to\infty}P_B(N,T)=P_B^U<\infty$.
- Property 3: For a given N, P_D increases with the increase of T.
  If $T1>T2$, then $P_D(N,T1)>P_D(N,T2), \forall\, N\geq T1$.
- Property 4: For a given T, P_D decreases with the increase of N and approaches zero.
  If $N1>N2$, then $P_D(N1,T)<P_D(N2,T), \forall\, T\leq N2$; and $\lim_{N\to\infty}P_D(N,T)=0$.
- Property 5: For a given N, P_L increases with the increase of T.
  If $T1>T2$, then $P_L(N,T1)>P_L(N,T2), \forall\, N\geq T1$.
- Property 6: For a given T, P_L increases with the increase of N and approaches an upper limit P_L^U.
  If $N1>N2$, then $P_L(N1,T)>P_L(N2,T), \forall\, T\leq N2$; and $\lim_{N\to\infty}P_L(N,T)=P_L^U<\infty$.

The mathematical proofs of the above properties can be found in [22]. From Properties 1–4, it is straightforward to obtain the following corollary.

Corollary 1: With the increase of N, the value of T that yields the same P_B is nondecreasing and the value of T that yields the same P_D is nondecreasing.

For any given N, we develop several computational modules as described in Figs. 1–5 to determine suboptimal solutions.

- Computational Module 1: Test if N is large enough to meet the constraints on P_B and P_D, i.e., if there exists any T for given N such that $P_B(N,T)\leq P_B^{\text{tar}}$ and $P_D(N,T)\leq P_D^{\text{tar}}$.
- Computational Module 2: Find the least integer $N_{\min}$ to satisfy both P_B and P_D constraints.
- Computational Module 3: Find $T_{\min}$ for the given N such that $P_B(N,T_{\min})=P_B^{\text{tar}}$.
- Computational Module 4: Find $T_{\max}$ for the given N such that $P_D(N,T_{\max})=P_D^{\text{tar}}$.
- Computational Module 5: Find $\hat{T}$ for the given N such that $P_L(N,\hat{T})=P_L^{\text{tar}}$.

$$\begin{aligned}P_B(N,T)&=\sum_{i=0}^{N}\pi_{N,T}(i)(1-P_{an}(i))\\&=\sum_{i=\lfloor T\rfloor}^{N}\pi_{N,T}(i)(1-P_{an}(i))\\&=\pi_{N,T}(\lfloor T\rfloor)(1-\beta)+\sum_{i=\lfloor T\rfloor+1}^{N}\pi_{N,T}(i)\\&=\frac{\frac{\rho^{\lfloor T\rfloor}}{\lfloor T\rfloor!}(1-\beta)+\sum\limits_{i=\lfloor T\rfloor+1}^{N}\frac{(\rho)^i}{i!}(\alpha\beta+(1-\alpha))(1-\alpha)^{i-\lfloor T\rfloor-1}}{\sum\limits_{i=0}^{\lfloor T\rfloor}\frac{\rho^i}{i!}+\sum\limits_{i=\lfloor T\rfloor+1}^{N}\frac{\rho^i}{i!}(\alpha\beta+(1-\alpha))(1-\alpha)^{i-\lfloor T\rfloor-1}}\end{aligned} \tag{15}$$

$$\begin{aligned}P_D(N,T)&=\sum_{i=0}^{N}\pi_{N,T}(i)(1-P_{ah}(i))=\pi_{N,T}(N)\\&=\frac{\frac{\rho^N}{N!}(\alpha\beta+(1-\alpha))(1-\alpha)^{N-\lfloor T\rfloor-1}}{\sum\limits_{i=0}^{\lfloor T\rfloor}\frac{\rho^i}{i!}+\sum\limits_{i=\lfloor T\rfloor+1}^{N}\frac{\rho^i}{i!}(\alpha\beta+(1-\alpha))(1-\alpha)^{i-\lfloor T\rfloor-1}}\end{aligned} \tag{16}$$

$$R_L(i)=\begin{cases}0, & 0\leq i\leq C\\ \frac{1}{iP_{\text{on}}}\sum\limits_{m=1}^{i-c}m\binom{i}{C+m}P_{\text{on}}^{C+m}(1-P_{\text{on}})^{i-C-m}, & i>C\end{cases} \tag{18}$$

```
/* Return 1 if P_B and P_D constraints can be both
satisfied and 0 else */
01 T = N
02 If (P_B(N,N) > P_B^tar)
03   RETURN 0
04 If (P_D(N,N) <= P_D^tar)
05   RETURN 1
06 U = N; L = 0
07 T = (U + L)/2
08 While ((P_B(N,T) > P_B^tar) XOR (P_D(N,T) > P_D^tar))
09   If (P_B(N,T) > P_B^tar)
10     L = T
11     T = (U + L)/2
12   Else If (P_D(N,T) > P_D^tar)
13     U = T
14     T = (U + L)/2
15 If ((P_B(N,T) <= P_B^tar) AND (P_D(N,T) <= P_D^tar))
16   RETURN 1
17 Else
18   RETURN 0
```

Fig. 1. Computational Module 1: Test if N is large enough to meet both constraints on P_B and P_D.

```
/* Return N_min such that P_B and P_D constraints
can be both satisfied */
01 INITIALIZE N = C + 1
02 If (P_D(N,N) <= P_D^tar)
03   RETURN N
04 U = N; L = 0
05 T = (U + L)/2
06 While ((P_B(N,T) > P_B^tar) XOR (P_D(N,T) > P_D^tar))
07   If (P_B(N,T) > P_B^tar)
08     L = T
09     T = (U + L)/2
10   Else If (P_D(N,T) > P_D^tar)
11     U = T
12     T = (U + L)/2
13 If ((P_B(N,T) <= P_B^tar) AND (P_D(N,T) <= P_D^tar))
14   RETURN N
15 Else
16   N = N + 1
17   GO TO 02
```

Fig. 2. Computational Module 2: Find the least integer $N_{\min}$ that satisfies both P_B and P_D constraints.

Computational Modules 2 and 4 and their proofs can also be found in [1]. Computational Modules 3 and 5 are developed similarly to computational module 4 using binary searching based on the properties given above. Note that computational modules 3, 4, or 5 are called only when the P_B, the P_D or the P_L constraint can be met for the given N, respectively. Based on these properties and computational modules, we are able to develop suboptimal solutions for each of the three formulated problems in the following.

1) Minimize P_L Subject to P_B and P_D Constraints: This problem could be reformulated as: to find the suboptimal CAC policy $\phi^* = (N^*, T^*)$ such that $P_L(N^*, T^*) = \min P_L(N,T)$ and $P_B(N^*, T^*) \le P_B^{\text{tar}}$, $P_D(N^*, T^*) \le P_D^{\text{tar}}$. The procedure is stated as follows.

Step 1) Let $N = C$ and use computational module 1 to test if there exists an T such that $P_B(N,T) \le P_B^{\text{tar}}$ and $P_D(N,T) \le P_D^{\text{tar}}$. If it is true, we obtain the minimum $P_L^* = 0$, and $N^* = C, T^* = T$. Otherwise, we proceed to the next step.

```
/* Return T_min under N such that P_B(N, T_min) = P_B^tar */
01 Resolution = 0.0001; MaxIter = 15
02 INITIALIZE Iter = 0
03 U = N; L = 0
04 T = (U + L)/2
05 While ((Iter < MaxIter) AND
         ((U - L) > Resolution))
06   If (P_B(N,T) > P_B^tar)
07     L = T
08     T = (U + L)/2
09   Else
10     U = T
11     T = (U + L)/2
12   Iter = Iter + 1
13   GO TO 05
14 RETURN T
```

Fig. 3. Computational Module 3: Find $T_{\min}$ for given N such that $P_B(N, T_{\min}) = P_B^{\text{tar}}$.

```
/* Return T_max under N such that P_D(N, T_max) = P_D^tar */
01 Resolution = 0.0001; MaxIter = 15
02 INITIALIZE Iter = 0
03 U = N; L = 0
04 T = (U + L)/2
05 If (P_D(N,N) <= P_D^tar)
06   RETURN N
07 While ((Iter < MaxIter) AND
         ((U - L) > Resolution))
08   If (P_D(N,T) > P_D^tar)
09     U = T
10     T = (U + L)/2
11   Else
12     L = T
13     T = (U + L)/2
14   Iter = Iter + 1
15   GO TO 07
16 RETURN T
```

Fig. 4. Computational Module 4: Find $T_{\max}$ for given N such that $P_D(N, T_{\max}) = P_D^{\text{tar}}$.

```
/* Return T̂ under N such that P_L(N, T̂) = P_L^tar */
01 Resolution = 0.0001; MaxIter = 15
02 INITIALIZE Iter = 0
03 U = N; L = 0
04 T = (U + L)/2
05 If (P_L(N,N) <= P_L^tar)
06   RETURN N
07 While ((Iter < MaxIter) AND
         ((U - L) > Resolution))
08   If (P_L(N,T) > P_L^tar)
09     U = T
10     T = (U + L)/2
11   Else
12     L = T
13     T = (U + L)/2
14   Iter = Iter + 1
15   GO TO 07
16 RETURN T
```

Fig. 5. Computational Module 5: Find $\hat{T}$ for given N such that $P_L(N, \hat{T}) = P_L^{\text{tar}}$.

Step 2) Find the minimum $N_{\min}(> C)$ such that both P_B and P_D constraints are satisfied using computational module 2.

Step 3) Let $N = N_{\min}$ and use computational module 3 to find the minimum $T_{\min}(N_{\min})$ such that both

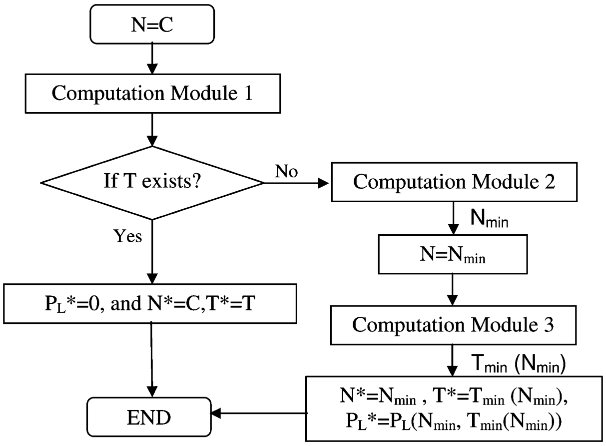

Fig. 6. Flowchart of the algorithm of minimizing P_L with P_B and P_D constraints.

P_B and P_D constraints are satisfied. Then, we have $N^* = N_{\min}$, $T^* = T_{\min}(N_{\min})$, and the corresponding minimum $P_L^* = P_L(N_{\min}, T_{\min}(N_{\min}))$.

Fig. 6 illustrates the flowchart of the procedure.

Proof: Let us start with $N = C$. If there exists an $T(0 \le T \le C)$ such that $P_B(C,T) \le P_B^{\text{tar}}$ and $P_D(C,T) \le P_D^{\text{tar}}$, P_L could be minimized as zero. Then, we obtain the solution and stop in Step 2). Note that there could be more than one T, the choice of the best T can be reduced to a connection-level optimization problem (including the optimization of P_B and P_D). Otherwise, P_B and P_D constraints cannot be met simultaneously without increasing N and causing $P_L > 0$. Using computational module 3, we can find a $T_{\min}(N)$ for any N by searching for T such that $P_B(N_{\min}, T) = P_B^{\text{tar}}$. According to Corollary 1, when $N > N_{\min}$, the $T_{\min}(N)$ that satisfies $P_B(N, T_{\min}(N)) = P_B^{\text{tar}}$ is nondecreasing, i.e., $T_{\min}(N) \ge T_{\min}(N_{\min})$. Then, with Properties 5 and 6, when $N > N_{\min}$, we have $P_L(N, T_{\min}(N)) > P_L(N_{\min}, T_{\min}(N_{\min}))$. Thus, the minimum P_L is equal to $P_L(N_{\min}, T_{\min}(N_{\min}))$.

2) Minimize P_B Subject to P_D and P_L Constraints: This problem could be reformulated as: to find the suboptimal CAC policy $\phi^* = (N^*, T^*)$ such that $P_B(N^*, T^*) = \min P_B(N,T)$ and $P_L(N^*, T^*) \le P_L^{\text{tar}}$, $P_D(N^*, T^*) \le P_D^{\text{tar}}$.

The optimization procedure is stated as follows.

Step 1) Set $N = C$ and $P_B^* = 1$ as initialization.
Step 2) Find $T_{\max}(N)$ that satisfies $P_D(N, T_{\max}(N)) = P_D^{\text{tar}}$ using computational module 4.
Step 3) Check if $P_L(N, T_{\max}(N)) \le P_L^{\text{tar}}$. If yes, proceed to Step 4); else, go to Step 6).
Step 4) Check if $P_B(N, T_{\max}(N)) \le P_B^*$. If yes, update $N^* = N$, $T^* = T_{\max}(N^*)$, and $P_B^* = P_B(N^*, T^*)$.
Step 5) Let $N = N + 1$ and go back to Step 2).
Step 6) Find the $T_{\min}(N)$ such that $P_B(N, T_{\min}(N)) = P_B^*$ using computational module 3.
Step 7) Check if $P_L(N, T_{\min}(N)) \le P_L^{\text{tar}}$. If yes, continue on Step 8); else, stop.
Step 8) Check if $P_D(N, T_{\min}(N)) \le P_D^{\text{tar}}$. If yes, continue on Step 9); else, go to Step 10).

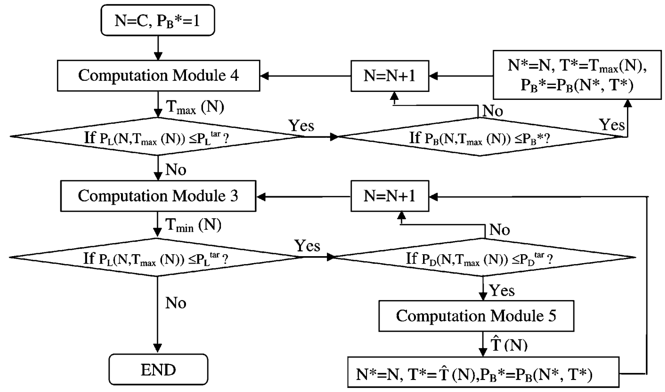

Fig. 7. Flowchart of the algorithm of minimizing P_B with P_D and P_L constraints.

Step 9) Search for $T = \hat{T}$ that satisfies $P_L(N, \hat{T}) = P_L^{\text{tar}}$ using Computational Module 5. Let $N^* = N$, $T^* = \hat{T}$, and $P_B^* = P_B(N, \hat{T})$.
Step 10) Let $N = N + 1$, and go back to Step 6) (until $P_L(N, T_{\min}(N)) > P_L^{\text{tar}}$).

Fig. 7 illustrates the flowchart of the optimization procedure.

Proof: Let us start with $N = C$. We can find $T_{\max}(N)$ that satisfies $P_D(N, T_{\max}(N)) = P_D^{\text{tar}}$ in Step 2). According to Properties 1 and 3, $T_{\max}(N)$ yields the minimum P_B under N. After the iteration of Step 5), we found $N_{\max} = N - 1$ and $T_{\max}(N-1)$ as the maximum numbers of N and T that satisfy $P_D(N, T_{\max}(N)) = P_D^{\text{tar}}$ and $P_L(N, T_{\max}(N)) \le P_L^{\text{tar}}$. Step 4) updates P_B^* as the minimum P_B for all $N \le N_{\max}$ such that both P_D and P_L constraints are satisfied.

Then, we determine if there exists smaller P_B that satisfies both P_D and P_L constraints when $N > N_{\max}$ using the following steps. In Step 7), if the P_L constraint is violated, for the current N, T has to be decreased from $T_{\min}(N)$ to meet the constraint according to Property 6. According to Property 2, we have the resulting $P_B \ge P_B^*$. It follows that, under the current N value, there is no solution to result in smaller P_B than P_B^* with the P_L constraint satisfied. Furthermore, we can show that increasing N can only results in higher P_L. According to Corollary 1, with the increase of N, $T_{\min}(N)$ is nondecreasing, i.e., $T_{\min}(N+1) \ge T_{\min}(N)$. According to Properties 5 and 6, we get $P_L(N+1, T_{\min}(N+1)) \ge P_L(N+1, T_{\min}(N)) > P_L(N, T_{\min}(N))$. Therefore, for given N, if $P_L(N, T_{\min}(N)) > P_L^{\text{tar}}$, $P_L(M, T_{\min}(M)) > P_L^{\text{tar}}$ for any $M > N$. Therefore, we can stop the iteration. The most updated N^* and T^* is the optimal solution with the minimum P_B^*.

In Step 8), if the P_D constraint is violated, T has to be decreased from $T_{\min}(N)$ to meet its constraint according to Properties 4. As discussed above, under the current N value, there is no solution to result in smaller P_B than P_B^* with both P_D and P_L constraints satisfied. Thus, we continue to examine the next N in Step 10) without updating the current optimal solution (N^*, T^*) and minimum P_B^*. Otherwise, in Step 9), since $P_L(N, T_{\min}(N)) \le P_L^{\text{tar}} = P_L(N, \hat{T})$, we have $\hat{T} \ge T_{\min}(N)$ with Property 6 and $P_B(N, \hat{T}) \le P_B^*$ with Property 2. Meanwhile, since $P_L(N, T_{\max}(N)) > P_L^{\text{tar}} = P_L(N, \hat{T})$, where

$T_{\max}(N)$ is such that $P_D(N, T_{\max}(N)) = P_D^{\text{tar}}$, we have $\hat{T} < T_{\max}(N)$. Thus, we get $P_D(N, \hat{T}) < P_D^{\text{tar}}$ according to Property 4. This shows that the updated $(N^*, T^*) = (N, \hat{T})$ is the optimal solution with the updated minimum P_B^* up to the current N, while both P_D and P_L constraints are satisfied.

Consider the scenario where the traffic load is so light with respect to the target P_L constraint that the upper bound $P_L^U < P_L^{\text{tar}}$. Then, even if every arriving connection is admitted, the average packet loss is still below the target. Theoretically, the CAC policy can minimize both P_D and P_B to zero by admitting all connection requests. This special case is not of our interest, but can be detected by stopping the iteration from Steps 1–5) when $P_L(N+1, T_{\max}(N+1)) - P_L(N, T_{\max}(N)) < \epsilon$, where ϵ is a very small value.

3) Minimize P_D Subject to P_B and P_L Constraints: This problem could be reformulated as: to find the suboptimal CAC policy $\phi^* = (N^*, T^*)$ such that $P_D(N^*, T^*) = \min P_D(N, T)$ and $P_L(N^*, T^*) \leq P_L^{\text{tar}}$, $P_B(N^*, T^*) \leq P_B^{\text{tar}}$. It is dependent on the parameters of the traffic load and target P_B and P_D. The optimal solution only exists when P_B^{tar} and P_L^{tar} are set properly for the traffic load conditions. Otherwise, there exist two trivial results.

- Case 1) The traffic load is so heavy that the target P_B and P_L constraints cannot be both satisfied.
- Case 2) The traffic load is so light that P_D could be minimized to zero, while both P_B and P_L constraints can be satisfied.

We use the following procedure to find the optimal solution.

- Step 1) Set $N = C$ in the initialization.
- Step 2) Let $T = N$, and test if $P_B(N, T) > P_B^{\text{tar}}$. If yes, let $N = N + 1$ and repeat Step 2) until $P_B(N, T) \leq P_B^{\text{tar}}$.
- Step 3) Let $N_{\min} = N$, and find $T_{\min}(N_{\min})$ under $N_{\min}$ such that $P_B(N_{\min}, T_{\min}(N_{\min})) = P_B^{\text{tar}}$ using Computational Module 3.
- Step 4) Check if $P_L(N_{\min}, T_{\min}(N_{\min})) > P_L^{\text{tar}}$. If yes, the result corresponds to Case 1); else, proceed to the next step.
- Step 5) Let $N^* = N_{\min}$ and $T^* = T_{\min}(N_{\min})$, and we have $P_D^* = P_D(N_{\min}, T_{\min}(N_{\min}))$, $P_L^* = P_L(N_{\min}, T_{\min}(N_{\min}))$.
- Step 6) Let $N = N + 1$, and find $T_{\min}(N)$ under N such that $P_B(N, T_{\min}(N)) = P_B^{\text{tar}}$ using Computational Module 3.
- Step 7) Check if $P_L(N, T_{\min}(N)) \leq P_L^{\text{tar}}$. If yes, go to Step 8). Otherwise, stop with the optimal solution (N^*, T^*).
- Step 8) Check if $P_D(N, T_{\min}(N)) \leq P_D^*$. If yes, update $N^* = N$, $T^* = T_{\min}(N)$, $P_D^* = P_D(N, T_{\min}(N))$, $P_L^* = P_L(N, T_{\min}(N))$. and then go to Step 9). Otherwise, go to Step 9) directly.
- Step 9) Check if $P_L(N, T_{\min}(N)) - P_L^* > \epsilon$, where ϵ is a very small value. If yes, go back to Step 6). Otherwise, the result is Case 2).

Fig. 8 shows the flowchart of the above procedure.

Proof: Steps 1–4) of the above procedure are used to detect if the result belongs to Case 1), i.e., to find if there exists an (N, T) pair such that $P_B(N, T) \leq P_B^{\text{tar}}$ and $P_L(N, T) \leq P_L^{\text{tar}}$. In Step 2), we find the minimum $N = N_{\min}$ that can meet the target P_B. According to Property 1, for any N, $N = T$ obtains the minimum $P_B(N, N)$. Then, in Step 3), we find the minimum $T = T_{\min}(N_{\min})$ under $N_{\min}$ that can meet the target P_B. According to Properties 1 and 2, the set of (N, T) that $P_B(N, T) \leq P_B^{\text{tar}}$ is a subset of $\{(N, T) | N \geq N_{\min}, T \geq T_{\min}(N_{\min})\}$. According to Properties 5 and 6, $P_L(N, T) \geq P_L(N_{\min}, T_{\min}(N_{\min}))$ for $(N, T) \in \{(N, T) | N \geq N_{\min}, T \geq T_{\min}(N_{\min})\}$. Therefore, if we find $P_L(N_{\min}, T_{\min}(N_{\min})) > P_L^{\text{tar}}$ in Step 4), then there is no $(N, T) \in \{(N, T) | N \geq N_{\min}, T \geq T_{\min}(N_{\min})\}$ that can satisfy $P_L(N, T) \leq P_L^{\text{tar}}$. In other words, there does not exist an (N, T) pair such that both the P_B and the P_L constraints can be satisfied, which is Case 1).

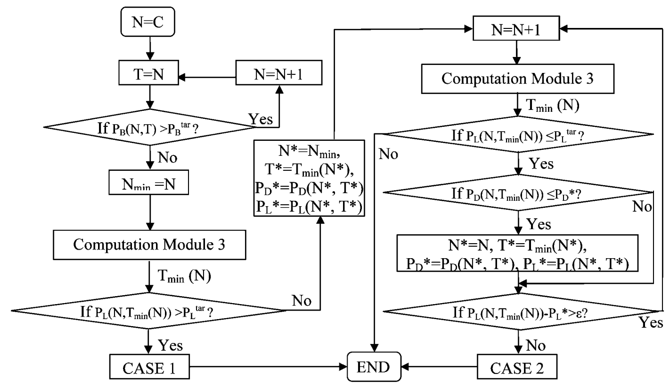

Fig. 8. Flowchart of the algorithm of minimizing P_D with P_B and P_L constraints.

Otherwise, if we can find $P_L(N_{\min}, T_{\min}(N_{\min})) \leq P_L^{\text{tar}}$ in Step 4), then there exists (N, T) that can meet both P_B and P_L constraints. Then, we continue the following procedure to find out the optimal solution and detect whether it belongs to Case 2). In Step 5), we find the optimal solution $(N^*, T^*) = (N_{\min}, T_{\min}(N_{\min}))$ that yields the minimum P_D, denoted by $P_D^* = P_D(N_{\min}, T_{\min}(N_{\min}))$, in the set of $N \leq N_{\min}$ that both P_B and P_L constraints are satisfied. Now, we only have to examine $N \geq N_{\min}$. In Step 6), when $N \geq N_{\min}$, $T_{\min}(N)$ results in the minimum P_D under N subject to the P_B^{tar} constraint. In Step 7), if $P_L(N, T_{\min}(N)) > P_L^{\text{tar}}$, there is no (N, T) for the current N and above that can satisfy both P_L and P_B constraints due to the same reason as that given in Step 4). Thus, the current (N^*, T^*) is the optimal one with minimum P_D^*. Otherwise, if necessary, the optimal solution is updated in Step 8) before proceeding to $N + 1$. The iteration stops either when $P_L(N, T_{\min}(N)) > P_L^{\text{tar}}$ in Step 7) with the optimal solution (N^*, T^*), or the $P_L(N, T_{\min}(N))$ converges to the upper limit in Step 9), which indicates the result of Case 2). In the latter case, the P_D could be minimized to zero by admitting all the handoff connection requests without violating both P_B and P_L constraints.

Once the optimal values of N and T are obtained, the corresponding CAC policy is predefined as follows. When the current load is less than T, any new or handoff connection request is admitted. When the current load is equal to the integer part of T (i.e., $\lfloor T \rfloor$), the handoff connection request is admitted, while the new connection request is admitted with probability of $T - \lfloor T \rfloor$. When the current load is greater than T and less than N, only handoff connection requests are admitted. When the current load reaches N, all connection requests are rejected.

When the traffic parameters change, the base station performs the appropriate procedure again to calculate a new set of optimal N and T, which defines a new stationary CAC policy.

III. Dynamic CAC Scheme for VBR Traffic

In Section II, we developed suboptimal stationary call admission control schemes for wireless VBR traffic. For mathematical tractability, the identically distributed on-off model was assumed for the VBR traffic. However, in real-world wireless multimedia networks, the VBR traffic is actually more complicated than the assumptions. Different multimedia applications may require a different amount of resource. Moreover, the traffic condition in terms of new/handoff call arrival and departure rates may be changing from time to time. Considering these practical aspects, the stationary optimal CAC schemes may not be efficient. Thus, dynamic schemes that take these factors into account are highly desirable. In this section, we develop a dynamic CAC scheme for more realistic VBR traffic.

The objective of our dynamic CAC scheme is to optimize the packet-level QoS under constraints on the connection-level QoS. It adjusts the two parameters N and T adaptively such that P_L can be minimized while keeping P_B and P_D under certain constraints P_B^{tar} and P_D^{tar}. Generally speaking, the more channels required to accommodate the aggregated peak data rates of multiplexed connections, the higher P_L will be for fixed channel capacity C. Therefore, in order to minimize P_L, the dynamic CAC scheme should achieve minimum N, while keeping P_B and P_D below their targets. Meanwhile, P_B and P_D provide a tradeoff upon the change of N or T. A higher N results in lower P_D but higher P_B under certain T, while a higher T results in lower P_B but higher P_D under certain N. Based on these observations, we develop a dynamic CAC scheme described in Fig. 9.

The base station measures both the P_B and P_D upon new call blocking and handoff dropping events. There are three cases when the value of N should be increased. First, if the measured $P_B > P_B^{\mathrm{tar}}$ when $T = N$, it is impossible to increase T further more without the increase of N to lower P_B. Second, if the measured P_B and P_D both exceed the targets, then either increasing or decreasing T cannot make both P_B and P_D lower than their target values. Third, if the measured $P_D > P_D^{\mathrm{tar}}$ when $T = 0$, then it is impossible to decrease T under the current N to lower P_D. In a practical wireless communication network, where the new call traffic is generally heavier than the handoff traffic, the last case rarely happens.

The timer setting detects when N is higher than necessity to meet the targets of P_B and P_D. Upon the timer expiration, it indicates that both P_B and P_D have been lower than their target values for a timer's period. Then, we decrease N to the minimum required value. With N that satisfies both P_B and P_D constraints under the current traffic condition, the dynamic change of T between 0 and N ensures $P_B \leq P_B^{\mathrm{tar}}$ and $P_D \leq P_D^{\mathrm{tar}}$. If the base station detects that $P_B > P_B^{\mathrm{tar}}$, it increases T to achieve lower P_B at the cost of higher P_D within its constraints. Otherwise, if the base station detects that $P_D > P_D^{\mathrm{tar}}$, it decreases T to achieve lower P_D at the cost of higher P_B within its constraint.

```
01 INITIALIZE N and T
02 SET TIMER
03 WAIT FOR CALL REQUEST ARRIVAL
04 If NEW CALL REQUEST ARRIVES
05    If (B_occupied + B_req) ≤ T
06       ADMIT NEW CALL WITH RATE B_req
07    Else
08       REJECT NEW CALL REQUEST
09       UPDATE P_B
10       If P_B ≥ P_B^tar
11          T = T + 1
12          If T > N
13             N = N + 1
14          RESET TIMER AND GO BACK TO 03
15 If HANDOFF CALL REQUEST ARRIVES
16    If (B_occupied + B_req) ≤ N
17       ADMIT HANDOFF CALL WITH RATE B_req
18    Else
19       REJECT HANDOFF REQUEST
20       UPDATE P_D
21       If P_D ≥ P_D^tar
22          If P_B ≥ P_B^tar OR T = 0
23             N = N + 1
24          Else
25             T = T − 1
26          RESET TIMER AND GO BACK TO 03
27 If TIMER EXPIRES AND N > C
28    N = N − 1
29    If T > N
30       T = N
31    RESET TIMER AND GO BACK TO 03
32 Else GO BACK TO 03
```

Fig. 9. Proposed dynamic CAC scheme for minimizing P_L of VBR multimedia traffic with both P_B and P_D constraints.

Unlike the stationary schemes, the dynamic CAC scheme does not assume any simplified traffic model. It is applicable to realistic multimedia traffic with heterogeneous data rates. Furthermore, the packet-level QoS metrics that can be minimized under the two connection-level QoS constraints is not limited to the congestion-caused packet loss probability. It is reasonable to assume that other packet-level QoS metrics, for example, delay or packet loss at other layers, have worse performance when the number of connections in the system increases. Therefore, the dynamic control process can also be applied as long as the QoS metrics follow similar properties as those discussed in Section II-C. Moreover, as a highly computational efficient real-time online control process, the proposed CAC scheme can adapt to dynamic changing traffic conditions. Therefore, it is more practical than the stationary ones.

IV. Experimental Results

In this section, experimental results are presented to demonstrate the performance of the stationary CAC schemes and dynamic CAC scheme as proposed in the previous two sections.

A. Stationary CAC Scheme

For stationary CAC schemes, the traffic pattern follows the model given in Section II-A with the following parameters.

- New call arrival rate: $\lambda_n = 40$ calls/s.
- Handoff call arrival rate: $\lambda_h = 10$ calls/s.
- Call departure rate: $\eta = 1$ call/s.
- On-state probability of each call: $P_{\mathrm{on}} = 0.25$.
- Cell capacity: $C = 20$.

TABLE I
RESULTS OF THE SUBOPTIMAL CAC SCHEMES FOR VBR TRAFFIC

Problem		Result				
min.	constraints	N	T	P_B	P_D	P_L
P_L	$P_B^{tar}=0.1$ $P_D^{tar}=0.01$	54	51.91	0.1	0.0026	4.74×10^{-4}
P_B	$P_D^{tar}=0.01$ $P_L^{tar}=10^{-6}$	41	34.45	0.42	0.00006	10^{-6}
P_D	$P_B^{tar}=0.3$ $P_L^{tar}=10^{-5}$	40	39.16	0.3	0.099	8.95×10^{-6}

Using the procedures described in Section II-C, we compute the optimal values of N and T of the stationary CAC policies and list them in Table I. The resulting packet-level QoS measurement P_L and connection-level QoS metrics P_B and P_D are also shown in Table I.

For the first example, the minimum of N to meet constraints $P_B^{\rm tar}=0.1$ and $P_D^{\rm tar}=0.01$ is $N_{\min}=54$. Next, we find the minimum T under N, $T_{\min}(N_{\min})=51.9052$, such that $P_B(N_{\min},T_{\min}(N_{\min})=0.1$. Therefore, the minimum P_L is $P_L(N_{\min},T_{\min}(N_{\min})=4.7359\times 10^{-4}$. The resulting $P_D(N_{\min},T_{\min}(N_{\min}))<P_D^{\rm tar}$.

For the second example, the maximum N such that $P_L(N,T_{\max}(N))\leq P_L^{\rm tar}=10^{-6}$ when $P_D(N,T_{\max}(N))=P_D^{\rm tar}=0.01$ is $N_{\max}=37$, and $T_{\max}(N_{\max})=34.1531$. The resulting $P_B(N_{\max},T_{\max}(N_{\max}))=0.4251$ is the minimum one when $N\leq 37$. When increasing N to $N=41$, $T=34.4481$ such that $P_L(N,T)=P_L^{\rm tar}$. The resulting $P_B(N,T)=0.4198$ is the minimum. Increasing N further to $N=42$ and above leads to $P_L(N,T)>P_L^{\rm tar}$ when $P_B(N,T)=0.4198$.

For the third example, the minimum N to satisfy the constraint $P_B^{\rm tar}=0.3$ is $N_{\min}=40$. The minimum T under $N_{\min}$ such that $P_B(N_{\min},T_{\min}(N_{\min}))=P_B^{\rm tar}$ is $T_{\min}(N_{\min})=39.1625$. The resulting $P_L=8.9542\times 10^{-6}<P_L^{\rm tar}=10^{-5}$, and $P_D(N_{\min},T_{\min}(N_{\min}))=0.0990$ is the minimum when $N\leq 40$. Increasing $N=41$ and above leads to $P_L(N,T)>P_L^{\rm tar}$ when both $P_B(N,T)\leq P_B$ and $P_D(N,T)\leq 0.0990$ are satisfied. If we change the target new call blocking probability $P_B^{\rm tar}=10^{-1}$, and the target packet loss probability $P_L^{\rm tar}=10^{-6}$, then there is no solution for which both P_B and P_L constraints can be met [i.e., Case 1)]. The minimum packet loss probability can be achieved is $P_L=3.5681\times 10^{-4}>P_L^{\rm tar}$ when $N=51$, $T=50.7553$, and the resulting $P_B=0.1000$, $P_D=0.0763$. On the other hand, if we change the target new call blocking probability $P_B^{\rm tar}=10^{-1}$, and the target packet loss probability $P_L^{\rm tar}=10^{-3}$, then the result belongs to Case 2), i.e., P_D could be minimized to zero by increasing N, and P_L reaches its upper limit when $N=51$, $T=50.7553$, $P_B=0.1000$, $P_D=0.0763$, and $P_L=3.5681\times 10^{-4}$.

B. Dynamic CAC Scheme

A more complicated multimedia traffic model is used in this subsection to test the dynamic CAC scheme. There is no closed-form solution available. The discrete-event simulator has been used to evaluate the performance of the proposed dynamic CAC algorithm when applied to multiple service types (i.e., video, audio, and voice, etc.) with VBR traffic at different peak rates.

TABLE II
TRAFFIC LOAD PARAMETER SETTINGS IN THE OPNET SIMULATION OF THE DYNAMIC CAC SCHEME FOR VBR TRAFFIC

Application type (i)	r_i (channels)	λ_{ni} (call/sec)	λ_{hi} (call/sec)	η_i (1/sec)
voice (1)	1	0.2	0.05	0.01
audio (2)	2	0.05	0.015	0.01
video (3)	4	0.025	0.005	0.01

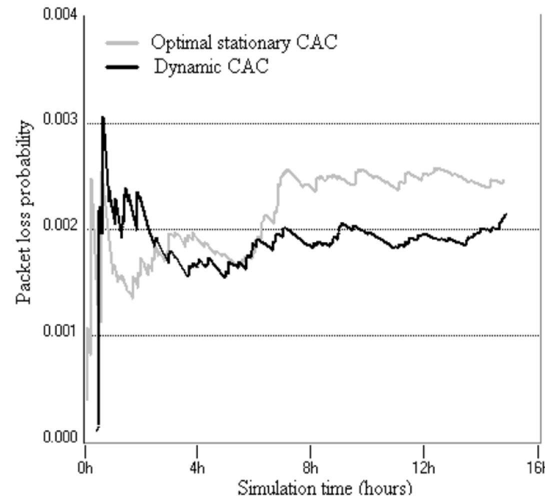

Fig. 10. Comparison of the packet loss probability of stationary and dynamic CAC schemes for VBR multimedia traffic.

A network model of a single cell with channel capacity $C=20$ was built with the OPNET simulator. Each channel has a data rate of 9600 bits/s. The multimedia VBR traffic model was built as an on–off model with configurable packet generation parameters listed below.

- The on-state time distribution: exponential with mean 1 s.
- The off-state time distribution: exponential with mean 3 s.
- The packet interarrival time distribution during on-state: exponential with mean 0.1 s.
- The packet size distribution: constant with mean 120 bytes.

The network traffic load configurations are shown in Table II. The r_i for each service type is the weighting factor of the peak rate during the ON period. It is used by the above VBR on–off model to scale the packet interarrival time generated.

The constraints of P_B and P_D are given as $P_B\leq P_B^{\rm tar}=0.1$ and $P_D\leq P_D^{\rm tar}=0.01$. For the comparison purpose, we found the optimal stationary CAC schemes with $N=56$ and $T=53.1$ by trial-and-error, such that the P_L is minimized, while P_B and P_D constraints are satisfied. Then, we compare P_L with that of the dynamic CAC scheme, as shown in Fig. 10. To illustrate the dynamic CAC scheme, the variation of N and T values are shown in Fig. 11. Simulation results for the two schemes are compared in Table III, where N and T are the averaged values over the simulation time. We can clearly see that the proposed dynamic CAC scheme for VBR multimedia traffic can achieve a lower packet loss probability than the stationary ones, while satisfying both P_B and the P_D constraints. Besides, more significant improvements of the dynamic scheme over the stationary ones include its applicability to realistic traffic and adaptivity to the changing traffic conditions.

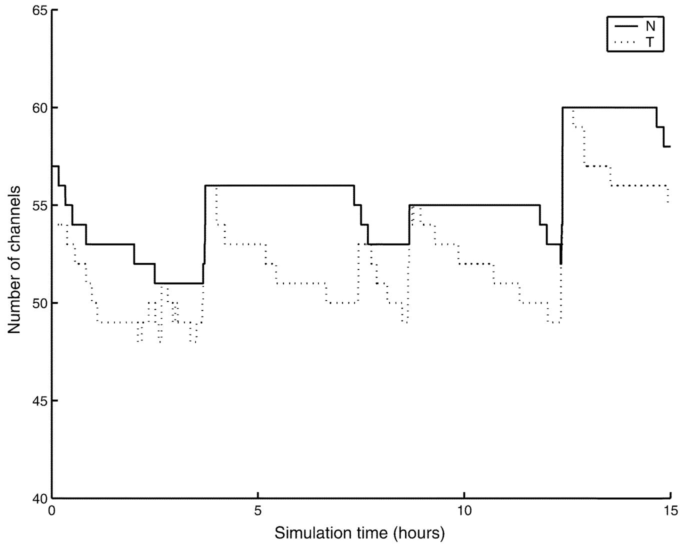

Fig. 11. Illustration of varying N and T for the dynamic CAC scheme for VBR multimedia traffic.

TABLE III
COMPARISON OF SIMULATION RESULTS BETWEEN STATIONARY AND DYNAMIC CAC SCHEMES FOR VBR TRAFFIC

CAC Scheme	P_B	P_D	P_L	N	T
Stationary	0.100	0.009	2.45×10^{-3}	56	53.1
Dynamic	0.094	0.010	2.13×10^{-3}	55.3	52.2

V. CONCLUSION AND FUTURE WORK

In this paper, we addressed the problem of providing connection-level QoS and packet-level QoS jointly for multimedia applications in the next-generation wireless networks. Specifically, the new call blocking probability (P_B), the handoff dropping probability (P_D), and the packet loss probability (P_L) for VBR traffic in wireless networks were analyzed based on a VBR traffic model. Three different joint optimization problems with different objectives and constraints were formulated. The suboptimal solution to each of the problems was developed. A dynamic CAC scheme was also proposed for the heterogeneous and varying multimedia traffic. Both stationary CAC and dynamic CAC schemes are computationally efficient for real-time implementation, and effectively provide joint packet-level QoS and connection-level QoS. Simulation results also showed that the dynamic CAC scheme achieves better performance than the stationary one when being applied to realistic multimedia traffic.

We are investigating the extension of the proposed schemes to more realistic VBR traffic models such as the self-similar VBR traffic model [23]. Future research directions include more practical considerations of the packet-level QoS metrics. For example, buffers can be employed to improve the packet-level QoS in terms of packet loss probability, but with increased delay. In that scenario, four QoS metrics need to be considered: two (new call blocking probability and handoff dropping probability) at connection-level and the other two (packet loss probability and delay) at packet-level. The joint QoS support involves both interlevel and intralevel tradeoffs in the following three aspects: the tradeoff between connection-level and packet-level QoS, the tradeoff between the two connection-level QoS metrics, and the tradeoff between the two packet-level QoS metrics. The call admission control, as well as scheduling schemes shall be investigated. In addition, the current work is based on the wireless system with a bandwidth-limited hard capacity. It is interesting and valuable to extend the current work to wireless systems with interference-limited soft capacity.

REFERENCES

[1] R. Ramjee, D. Towsley, and R. Nagarajan, "On optimal call admission control in cellular networks," *Wireless Netw.*, vol. 3, no. 1, pp. 29–41, May 1997.

[2] M. Naghshineh and S. Schwartz, "Distributed call admission control in mobile/wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 14, no. 4, pp. 711–717, May 1996.

[3] S. Wu, K. Y. M. Wong, and B. Li, "A new distributed and dynamic call admission policy for mobile wireless networks with QoS guarantee," in *Proc. 9th IEEE Int. Symp. Pers., Indoor, Mobile Radio Commun.*, vol. 1, Sep. 1998, pp. 260–264.

[4] J. M. Peha and A. Sutivong, "Admission control algorithms for cellular systems," *Wireless Netw.*, vol. 7, no. 2, pp. 117–125, Apr. 2001.

[5] D. Levine, I. Akyildiz, and M. Naghshineh, "A resource estimation and call admission algorithm for wireless multimedia networks using the shadow cluster concept," *IEEE/ACM Trans. Netw.*, vol. 5, no. 1, pp. 1–12, Feb. 1997.

[6] T. Kwon, Y. Choi, C. Bisdikian, and M. Naghshineh, "Measurement-based call admission control for adaptive multimedia in wireless/mobile networks," in *Proc. IEEE Wireless Commun. Netw. Conf.*, vol. 2, May 1999, pp. 540–544.

[7] J. M. Capone and I. Stavrakakis, "Delivering diverse delay/dropping QoS requirements in a TDMA environment," in *Proc. ACM MobiCom*, vol. 1, Sep. 1997, pp. 110–119.

[8] N. R. Figueira and J. Pasquale, "Providing quality of service for wireless links: Wireless/wired networks," *IEEE Pers. Commun.*, vol. 6, no. 5, pp. 42–51, Oct. 1998.

[9] C. Oliveira, J. B. Kim, and T. Suda, "An adaptive bandwidth reservation scheme for high-speed multimedia wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 6, pp. 858–874, Aug. 1998.

[10] 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.

[11] A. K. Talukdar, B. R. Badrinath, and A. Acharya, "MRSVP: A resource reservation protocol for an integrated services network with mobile hosts," *Wireless Netw.*, vol. 7, no. 1, pp. 5–19, Jan. 2001.

[12] R. Braden, D. Clark, and S. Shenker, "Integrated service in the Internet architecture: An overview," IETF, RFC1633, 1994.

[13] W. Yang and E. Geraniotis, "Admission policies for integrated voice and data traffic in CDMA packet radio networks," *IEEE J. Sel. Areas Commun.*, vol. 12, no. 4, pp. 654–664, May 1994.

[14] T. Liu and J. A. Silvester, "Joint admission/congestion control for wireless CDMA systems supporting integrated services," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 6, pp. 845–857, Aug. 1998.

[15] C. Cho, S. M. Shin, and D. K. Sung, "Interference-based channel assignment for DS-CDMA cellular systems," *IEEE Trans. Veh. Technol.*, vol. 48, no. 1, pp. 233–239, Jan. 1999.

[16] W. S. Jeon and D. G. Jeong, "Call admission control for CDMA mobile communications systems supporting multimedia services," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 649–659, Oct. 2002.

[17] H. Liu and M. E. Zarki, "Transmission of video telephony images over wireless channels," *Wireless Netw.*, vol. 2, no. 3, pp. 219–228, Aug. 1996.

[18] S. Kaiser, "OFDM code-division multiplexing in fading channels," *IEEE Trans. Commun.*, vol. 50, no. 8, pp. 1266–1273, Aug. 2002.

[19] S. Siwamogsatham and M. P. Fitz, "Robust space–time codes for correlated Rayleigh-fading channels," *IEEE Trans. Signal Process.*, vol. 50, pp. 2408–2416, Oct. 2002.

[20] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A media access protocol for wireless LANs," in *Proc. Conf. Commun. Arch., Protocols, Applicat.*, vol. 24, London, U.K., Oct. 1994, pp. 212–225.

[21] J. J. Garcia-Luna-Aceves and A. Tzamaloukas, "Receiver-initiated collision avoidance in wireless networks," *Wireless Netw.*, vol. 8, no. 2/3, pp. 249–263, Mar. 2002.

[22] L. Huang, "Call admission control and resource allocation for quality of service support in wireless multimedia networks," Ph.D. dissertation, Univ. Southern California, Los Angeles, CA, 2003.

[23] M. W. Garrett and W. Willinger, "Analysis, modeling and generation of self-similar VBR video traffic," in *Proc. Conf. Commun. Arch., Protocols, Applicat.*, vol. 24, Aug. 1994, pp. 269–280.

Lei Huang (M'04) received the B.S. and M.S. degrees in electrical engineering from Beijing University of Posts and Telecommunications, Beijing, China, in 1993 and 1996, respectively, and the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, in 2003.

Since August 2003, she has been an Assistant Professor in the Department of Electrical Engineering and Computer Science, Loyola Marymount University, Los Angeles. Her research interests include digital image and video coding, multimedia applications in wired and wireless networks, quality-of-service, and network security.

C.-C. Jay Kuo (S'83–M'86–SM'92–F'99) received the B.S. degree in electrical engineering from the National Taiwan University, Taipei, in 1980 and the M.S. and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1985 and 1987, respectively.

He was Computational and Applied Mathematics (CAM) Research Assistant Professor in the Department of Mathematics, University of California, Los Angeles, from October 1987 to December 1988. Since January 1989, he has been with the Department of Electrical Engineering-Systems and the Signal and Image Processing Institute, University of Southern California, Los Angeles, where he currently has a joint appointment as Professor of Electrical Engineering and Mathematics. He has guided about 60 students to their Ph.D. degrees and supervised 15 Postdoctoral Research Fellows. He is coauthor of seven books and more than 700 technical publications in international conferences and journals. His research interests are in the areas of digital signal and image processing, audio and video coding, multimedia communication technologies and delivery protocols, and embedded system design.

Dr. Kuo is a Fellow of the International Society for Optical Engineers (SPIE) and a member of the Society for Industrial and Applied Mathematics (SIAM), and the Association for Computing Machinery (ACM). He received the National Science Foundation Young Investigator Award (NYI) and Presidential Faculty Fellow (PFF) Award in 1992 and 1993, respectively. He is Editor-in-Chief for the *Journal of Visual Communication and Image Representation*, Associate Editor for the IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING, and Editor for the *Journal of Information Science and Engineering* and the *EURASIP Journal of Applied Signal Processing*. He is also on the Editorial Board of the *IEEE Signal Processing Magazine*. He served as Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING from 1995 to 1998 and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 1995 to 1997.