QoS Provisioning Dynamic Connection-Admission Control for Multimedia Wireless Networks Using a Hopfield Neural Network

Chang Wook Ahn, *Student Member, IEEE,* and R. S. Ramakrishna, *Senior Member, IEEE*

***Abstract*—This paper presents a quality-of-service (QoS) provisioning dynamic connection-admission control (CAC) algorithm for multimedia wireless networks. A multimedia connection consists of several substreams (i.e., service classes), each of which presets a range of feasible QoS levels (e.g., data rates). The proposed algorithm is mainly devoted to finding the best possible QoS levels for all the connections (i.e., QoS vector) that maximize resource utilization by fairly distributing wireless resources among the connections while maximizing the statistical multiplexing gain (i.e., minimizing the blocking and dropping probabilities). In the case of congestion (overload), the algorithm uniformly degrades the QoS levels of the existing connections (but only slightly) in order to spare some resources for serving new or handoff connections, thereby naturally minimizing the blocking and dropping probabilities (it amounts to maximizing the statistical multiplexing gain). The algorithm employs a Hopfield neural network (HNN) for finding a QoS vector. The problem itself is formulated as a multi-objective optimization problem. Hardware-based HNN exhibits high (computational) speed that permits real time running of the CAC algorithm. Simulation results show that the algorithm can maximize resource utilization and maintain fairness in resource sharing, while maximizing the statistical multiplexing gain in providing acceptable service grades. Furthermore, the results are relatively insensitive to handoff rates.**

***Index Terms*—Dynamic connection-admission control (CAC), Hopfield neural network (HNN), multimedia services, quality of service (QoS), wireless networks.**

I. Introduction

IN WIRELESS networks, the demand for multimedia services, including different classes of service (i.e., substreams) with widely different traffic characteristics, has increased. Since the profiles and requirements of the services are quite dynamic in nature, the quality of service (QoS) expected from these services also differs rather widely [1]–[14]. It is not easy to properly allocate resources to the multimedia connections because the bandwidth requirements are highly unpredictable. Thus, achieving high utilization of the system and, with fair resource allocation at that, is difficult. In addition, it tends to decrease the cell size for providing more capacity in order to accommodate more mobile/wireless connections [1]–[4]. It is more difficult to guarantee negotiated QoS grades because the reduction of cell size tends to induce more frequent handoffs. Therefore, wireless networks for multimedia services must incorporate an efficient connection-admission control (CAC) strategy, not only to maximize system utilization but also to fairly distribute resources among different connections while maximizing the statistical multiplexing gain (i.e., no blocking of new connections and no dropping of handoff connections as far as possible) [5]–[14]. In addition, the promised QoS levels for all the connections should be guaranteed. Such a CAC algorithm requires that an appropriate multi-objective optimization problem involving an enormous search space be solved (see Sections II-A and III-A). In addition to the dual goals of higher resource utilization and fairness in resource sharing, the CAC has to operate in real time; hence, the computational burden must be consistent with the available processing capacity [12], [13].

There are two well known approaches in this regard: the static and the dynamic approaches. The static approach [2]–[4] typically tries to find an acceptable solution to deal with new and handoff connections so as to guarantee the negotiated QoS (e.g., handoff dropping probability) in nonmultimedia wireless networks. The Factional Guard Channel method is considered to be an optimal scheme [2], [3]. Most recently, many researchers have been focusing on the dynamic approach [5]–[14], which generally tries to maximize resource utilization while guaranteeing QoS contract in multimedia wireless networks. In general, the negotiated QoS level remains fixed during services once a connection is admitted to the network (viz., nonadaptive multimedia services). There is an attractive service scenario in multimedia wireless networks that is worth investigating: adaptive multimedia services [7], [8], [14]. Each connection generally declares a predefined range of feasible QoS levels (described in a user-defined profile (UDP) [14]) and new or handoff connections can be admitted to the networks even if the remaining resources are not sufficient to satisfy their highest QoS guarantees (i.e., maximum data rate). Hence, compromising the QoS levels for the existing connections is inevitable in order to set apart some bandwidth for the new or handoff connections. Therefore, the aim of dynamic approach is to find the best possible QoS levels (i.e., QoS vector) so as to satisfy the design objectives (viz., maximizing resource utilization and fairly distributing the resources while maximizing the statistical multiplexing gain with guaranteeing the promised QoS levels). As mentioned before, it naturally falls within the purview of a multi-objective optimization problem.

Manuscript received January 15, 2003; revised May 1, 2003 and August 28, 2003. This work was supported in part by the Korean Ministry of Education under the BK21 Program.

The authors are with the Department of Information and Communications, Kwang-Ju Institute of Science and Technology (K-JIST), Gwangju 500-712, Korea (e-mail: cwan@kjist.ac.kr; rsr@kjist.ac.kr).

Digital Object Identifier 10.1109/TVT.2003.822000

In order to pursue the goals, an efficient scheduler for dynamically allocating resources is necessary. In wireless networks, the medium access control (MAC) protocol itself should play a central role in dynamically scheduling resources because the scheduler does not have direct access to the buffer state information, such as instantaneous queue length and cell arrival rate, due to the spatially distributed nature of mobile connections [15]. The performance of the scheduler depends mainly on how quickly and accurately such information can be transmitted from mobile connections to the scheduler. Numerous efficient MAC protocols in wireless networks have been developed in this regard [15]–[17], but this issue is beyond the scope of this paper.

We can classify CAC schemes into two strategies: classical and intelligent. Many classical schemes have been proposed [2], [3], [5]–[11]. Ramjee *et al.* [2] and Ho and Lea [3] proposed a special guard channel policy in order to effectively reserve a nonintegral number of guard channels for handoff connections. Epstein and Schwartz [5] extended the guard channel scheme (of the static CAC) to multimedia services, but the reservation partitions are still static and are not adapted to the offered load. Kwon *et al.* [6] proposed the admission-control and bandwidth-reallocation algorithms for multimedia services by using a type of greedy algorithm. Xiao *et al.* [7] developed a model-based CAC algorithm for adaptive multimedia services by adopting the layered (i.e., hierarchical) coding scheme (e.g., MPEG, H.263) and the semi-Markov decision process (SMDP). The layered coding scheme is used for dynamically changing the demanded bandwidth in accordance with the network condition. For instance, all MPEG video frames (i.e., I, P, and B frames) are transmitted when enough bandwidth is available. Otherwise, only I and P frames or only the I frames are transmitted as per the available bandwidth [7], [14]. Xiao and Chen [8] also proposed a proportional degradation mechanism based on QoS parameters [i.e., degradation ratio (DR) and degradation degree (DD)] for multiple classes of adaptive multimedia services. Misic *et al.* [9] developed a multimedia reservation-based admission algorithm that ensures the provision of adequate QoS to connections in the networks as defined by the dropping probability. Choi and Shin [10] developed a (mathematical) mobility model and proposed a CAC algorithm based on their model, which allows for connections of different bandwidths under the assumption of a single traffic class. Epstein and Schwartz [11] proposed two new predictive admission-control algorithms for multiclass traffic in wireless networks. They extended the notion of QoS for multimedia services in terms of maximum dropping probability (independent of the offered load) and predefined blocking probabilities for the different traffic classes.

CAC algorithms employing fuzzy logic, neural networks, and evolutionary computation (e.g., genetic algorithms) are included in the intelligent strategy [4], [12]–[14]. Such approaches are deemed to be feasible and practical because the classical strategy usually suffers from estimation error induced by modeling and approximation error owing to real-time requirements [12]. Furthermore, they are considered to be quite efficient heuristic methods for tackling complex optimization problems. Lázaro and Girma [4] developed an adaptive guard channel policy for nonmultimedia services by employing a Hopfiled neural network (HNN) to maintain an optimum balance between handoff connection dropping and new connection-blocking probabilities. Cheng *et al.* [12] proposed a neural-network based fuzzy admission-control algorithm for multimedia high-speed networks that guarantees a predefined QoS and promises a high resource utilization. Ren and Ramamurthy [13] developed an admission-control and bandwidth-allocation algorithm by applying fuzzy logic control to online measurement of the multiclass traffics in the wireless network, for achieving higher utilization while guaranteeing QoS requirements. Sherif *et al.* [14] employed genetic algorithms for developing an efficient dynamic CAC algorithm for (adaptive) multimedia services in wireless asynchronous transfer mode (ATM) networks. It could maintain a high resource utilization while minimizing the connection-blocking probability, at the expense of temporarily and slightly degrading the QoS levels perceived by the existing connections.

Although the genetic-algorithmic approach [14] can provide a good solution (i.e., QoS vector) to the optimization problem, it has a serious problem associated with operating in real time. The fuzzy-logic-based approach [13] is excellent in dealing with real-world imprecision and has a greater ability to adapt itself to dynamic and imprecise environments. However, it lacks the knowledge needed to automatically construct its rule structure and membership functions so as to get a good solution [12]. The neural-network approach [4], [12] [especially hardware implementation such as the field-programmable gate array (FPGA) chip], on the other hand, provides feasible solutions to complex optimization problems within a very short time (i.e., real time) [18], [19]. The hardware size or amount of power consumption of the neural network is not significant because the CAC algorithm runs at base/central stations. Thus, neural networks are considered to be promising candidates for designing (dynamic) CAC algorithm for multimedia services.

In this paper, we propose a QoS provisioning dynamic CAC algorithm for adaptive multimedia services in wireless networks. The objectives include maximization of the utility factor of scarce wireless resources, minimization of the blocking and dropping probabilities (i.e., maximize the statistical multiplexing gain) in real time, and provision of fair distribution of resources among the connections for providing acceptable service grades. We employ the HNN as a means to effectively explore the large search space (of the problem of finding the optimal QoS vector). The most attractive feature of HNN is that it is a recurrent network that operates in an unsupervised mode, requiring no training. The layered encoded traffic sources provide the framework for supporting the dynamic resource allocation by reducing data rate (i.e., degrading the QoS level). Hence, the proposed algorithm will concentrate on carefully degrading the QoS levels of the existing connections in order to spare some bandwidth for new or handoff connections. The solution to dynamic CAC comes within the purview of computing the best QoS levels for all the connections amid the large search space (i.e., possible combinations). The advantage of the hardware-based HNN is its ability to work in real time (e.g., a few microseconds). Simulation studies bring out the effectiveness of the proposed algorithm with regard to resource utilization, statistical multiplexing gain, and fair allocation of resources.

The rest of the paper is organized as follows. The motivation for considering HNN in this regard is described in Section II. The HNN structure is also in this section. Section III describes the proposed algorithm and simulation results that bring forth the performance of the proposed algorithm are presented in Section IV. Finally, the paper concludes in Section V.

II. HNN Model

A. Motivation

In addition to achieving high utilization of the system and low blocking and dropping probabilities, a real-time capability is one of the most important factors for developing useful CAC algorithms. In solving a dynamic CAC problem formulated as a multi-objective optimization problem (Section III-A), the QoS is quite important because it directly affects the performance of the system. When the search space of a certain problem is not very large, an optimal solution can be found by mechanically computing every combination by brute force. Unfortunately, the search space is generally too large for an exhaustive search. For instance, a multimedia connection consists of three kinds of service (e.g., video, voice, and data substreams) and each service is allowed for servicing three QoS levels (e.g., high, medium, and low). Even if there are five multimedia connections in the system, the search space becomes quite unmanageable (i.e., $(3^3)^5 \approx 1.435 \times 10^7$ combinations) [14]. As described in Section I, HNN is a promising candidate for such computations (in the dynamic CAC problem) because their hardware approach offers acceptable solutions quite quickly (e.g., $10 \sim 20$ μs) [19]. Moreover, it does not use any training sequence because neurons are evolving into their stable states in an unsupervised manner.

B. Dynamics of HNN

The use of neural networks for solving optimization problems was initiated by Hopfield and Tank [20]. They demonstrated the computational power of neural networks by applying their model to the traveling salesman problem (TSP). Since then, many investigators have concentrated on applying the Hopfield model to various optimization problems [18], [19], [21]–[23]. In the conventional HNN, each neuron is modeled as a nonlinear device (i.e., operational amplifier) with a sigmoid f_i, monotonically increasing function defined by the logistic function

$$V_i = f_i(U_i) = \frac{1}{1 + e^{-\alpha_i U_i}} \tag{1}$$

where U_i is the input of ith neuron, V_i taking on any value between 0 and 1 is the output of ith neuron, and α_i is the gain of the amplifier on the ith neuron.

Each neuron receives resistive connections (modeling the biological synaptic connection) from other neurons and these connections can be fully described by the interconnection matrix $\mathbf{T} = [T_{ij}]$. Here, T_{ij} is the interconnection weight from the jth neuron to the ith neuron. Each neuron also receives an input bias current I_i, which is the only user-adjustable parameter. The dynamics of the HNN are represented by

$$\frac{dU_i}{dt} = \sum_{k=1}^{L} T_{ik} V_k - \frac{U_i}{\tau} + I_i \tag{2}$$

[18], [20], [24] where τ is the time constant of the circuit and L is the number of neurons.

Hopfield [25] proved that for a symmetric interconnection matrix $\mathbf{T}$ and sufficiently high gains of the amplifiers, neurons evolve by gradient descent of the quadratic energy function E, as follows:

$$E = -\frac{1}{2} \sum_{i=1}^{L} \sum_{k=1}^{L} T_{ik} V_i V_k - \sum_{i=1}^{L} I_i V_i. \tag{3}$$

Using (3), (2) may be rewritten as

$$\frac{dU_i}{dt} = -\frac{U_i}{\tau} - \frac{\partial E}{\partial V_i}. \tag{4}$$

It is noted that the minima of the energy function (3) occur at the 2^L corners inside the L-dimensional hypercube (i.e., search space) defined on $V_i \in \{0, 1\}$ [18], [25]. In the previous example (i.e., five multimedia connections, three services, and three QoS levels per connection), the search space of the problem is far larger than that of the algebraic domain [i.e., $(3^3)^5 \approx 1.435 \times 10^7$], because the minima occur at $2^{3^3 \times 5} \approx 4.356 \times 10^{40}$ corners in the search space [i.e., $(3^3 \times 5)$-dimensional hypercube] when the HNN is employed. In spite of this disadvantage (the extremely large search space), the HNN is still an efficient strategy for solving such problems whenever its hardware approach is employed.

III. Proposed QoS Provisioning Dynamic CAC Algorithm

A. Problem Definition

As described in Section I, the system environment akin to that of [7], [8] and [14] is considered. It is assumed that a multimedia connection consists of three classes of service (substreams)—video, voice, and data. Each service can be served at three QoS levels (i.e., high, medium, and low qualities) through various coding schemes for real-time services (e.g., MPEG or H.263 for video encoding [7], [14]and embedded ADPCM for dynamic voice encoding [26]) and the data-rate-reduction scheme for nonreal-time services such as the data substream. The information (i.e., bandwidth according to QoS levels) attached with control messages is gathered at the base station (BS) in the signaling phase. This process is triggered by new/handoff arrivals or departures and also by requests for readjusting QoS contracts due to the dynamic nature of the adaptive multimedia services. We consider N connections in the system and define the QoS vector $\mathbf{Q} = (Q_1, Q_2, \cdots, Q_N)$ where Q_i presents the QoS level for connection i and $Q_i = \{j | j \in Z^+, 1 \leq j \leq M\}$. Here, Z^+ and M denote the set of positive integers and the number of possible QoS levels, respectively. M is 3^3 in the above example. The first 1 and the last M (i.e., 27) indices denote the "best" and the "worst" QoS levels, respectively.

Here, the index "1" indicates the service offering high-resolution video, high-quality audio, and high-speed data, while the index "27" refers to the service providing low-resolution video, low-quality audio, and low-speed data [14]. We also define the bandwidth vector $\mathbf{B} = (B_1^{Q_1}, B_2^{Q_2}, \cdots, B_N^{Q_N})$ where $B_i^{Q_i}$ indicates the required bandwidth of connection i in order to provide the QoS level (Q_i).

In the case of a light load, the system can serve all the connections with their best QoS levels (i.e., for all i, $Q_i = 1$) without any calculation for dynamically allocating wireless resources. Therefore, we only have to deal with the overload case. In this case, the design objective is not only to maximize the system utilization, but also to maintain fairness in resource allocation while maximizing the statistical multiplexing gain (i.e., minimizing the blocking and the dropping probabilities). Given a total bandwidth B_T and a total of N multimedia connections, the proposed algorithm tries to find the best QoS vector $\mathbf{Q}^*$ (i.e., optimal solution) so as to minimize

$$f_1 = \sum_{i=1}^{N} \left(B_T \frac{B_i^1}{\sum_{j=1}^{N} B_j^1} - B_i^{Q_i} \right)^2 \tag{5a}$$

and

$$f_2 = \left(B_T - \sum_{i=1}^{N} B_i^{Q_i} \right)^2 \tag{5b}$$

subject to

$$\sum_{i=1}^{N} B_i^{Q_i} \leq B_T \tag{5c}$$

where B_i^1 the bandwidth (of connection i) that offers the best QoS level $(Q_i = 1)$.

The first objective (5a) nudges the system toward fair distribution of resources among different connections. The second objective (5b) drives the system toward higher resource utilization. The constraint (5c) ensures that the sum of allocated bandwidths does not exceed the total available capacity (B_T). Developing a dynamic CAC algorithm for multimedia services corresponds to solving this multi-objective optimization problem.

It implies that (5a) is related to proportional fairness concept [27], [28] in bandwidth sharing, which comes from recent economic theory. Proportional fairness maximizes an objective function representing overall utilization of all the connections while keeping the capacity constraint. Therefore, fairness in shared wireless channels is defined by

$$\begin{aligned} \text{fairness} &= 1 - \frac{1}{N} \sum_{i=1}^{N} \left| \frac{\widetilde{B}_i - B_i^{Q_i}}{\widetilde{B}_i} \right| \\ &= 1 - \frac{1}{N} \sum_{i=1}^{N} \left| 1 - \frac{B_i^{Q_i}}{\widetilde{B}_i} \right| \end{aligned} \tag{6}$$

in this paper. Here, $\widetilde{B}_i = B_T(B_i^1 / \sum_{j=1}^{N} B_j^1)$. It indicates a fraction of the available bandwidth, which is proportional to the corresponding maximum requirement. Hence, fairness (6) is related to the ratio of the allocated bandwidth to a fraction of the available bandwidth. In this paper, the value 1 indicates that a bandwidth allocation is completely fair.

B. Problem Formulation in Terms of HNN

To formulate the optimization problem in terms of HNN, the computational network requires $N \times M$ neurons where N and M represent the number of connections and the number of QoS levels, respectively. The output values of neurons, denoted by $\mathbf{V} = [V_{ij}]$, indicate a QoS assignment matrix for each connection. Each output V_{ij}, defined as in (7), characterizes a neuron at location (i, j)

$$V_{ij} = \begin{cases} 1, & \text{if } Q_i = j \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

There is a QoS indicator matrix $\Psi = [\psi_{ij}]$ defined as

$$\psi_{ij} = \begin{cases} 1, & \text{if } Q_i \dashv j \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

Here, $Q_i \dashv j$ indicates that the QoS level j is usable to the connection i.

This may be used for serving different grades of service to which the users can subscribe. For instance, a subscriber to a premium service will be allocated the QoS indicator with the uppermost range of QoS levels, whereas a subscriber to an economy service will have the QoS indicator that is lower than the premium one [14]. The premium service may allow high quality on video service, high-to-medium quality on voice service, and all three qualities on data service. In that case, the subscriber i will be assigned to the QoS indicator vector $\Psi_i = \{\delta_{kj} | 1 \leq k \leq M$ (e.g., 27), $1 \leq j \leq 6$, $k, j \in Z^+\}$, where δ_{kj} is the Kronecker delta function.

There is a cost C_{ij} associated with connection i corresponding to the QoS level j. The costs are specified by the cost matrix $\mathbf{C} = [C_{ij}]$, where C_{ij} denotes the cost related to the fairness (6) in resource allocation among the connections. In this context, the cost C_{ij} is defined as

$$C_{ij} = \frac{\left\{ B_T \left(\frac{B_i^1}{\sum_{k=1}^{N} B_k^1} \right) - B_i^j \right\}^2}{\max\limits_{\substack{\forall x \in \{1, \cdots, N\} \\ \forall y \in \{1, \cdots, M\}}} \left\{ B_T \left(\frac{B_x^1}{\sum_{k=1}^{N} B_k^1} \right) - B_x^y \right\}^2} \tag{9}$$

where B_i^j indicates the bandwidth of connection i with QoS level j $(Q_i = j)$.

In (9), the numerator is exactly the same as the right-hand side of (5a), which indicates a fraction of the available bandwidth [which is conceptually identical to (6)]. The denominator is a normalized constant needed to make the cost lie between zero and one. Using the above definitions, the HNN energy function for satisfying the objectives and the constraint described in (5) can be written as

$$\begin{aligned} E = &\frac{\mu_1}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} C_{ij} V_{ij} + \frac{\eta^{\zeta} \mu_2}{2} \left| 1 - \sum_{i=1}^{N} \sum_{j=1}^{M} \frac{B_i^j}{B_T} V_{ij} \right| \\ &+ \frac{\mu_3}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} \psi_{ij} V_{ij} + \frac{\mu_4}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} V_{ij}(1 - V_{ij}) \\ &+ \frac{\mu_5}{2} \sum_{i=1}^{N} \left(1 - \sum_{j=1}^{M} V_{ij} \right)^2 \end{aligned} \tag{10}$$

where $\zeta = u(\sum_{i=1}^{N} \sum_{j=1}^{M} (B_i^j / B_T) V_{ij} - 1)$ and $u(\bullet)$ is a unit step function.

Minimization of the proposed energy function drives each neuron of the HNN into its stable state, thereby providing a feasible QoS vector. In (10), the μ_1 term minimizes the total cost of a QoS vector, thereby fairly and maximally allocating resources to different connections. As explained before, it directly reflects on (5a), which is physically identical to the fairness advocated by (6). The μ_2 term involves both the objective (5b) and the constraint (5c) satisfaction. This is because minimization of the energy function E draws $\sum_{i=1}^{N}\sum_{j=1}^{M}(B_i^j/B_T)V_{ij}$ to 1. In other words, since it is striving to push $\sum_{i=1}^{N}\sum_{j=1}^{M}B_i^jV_{ij}$ (i.e., it denotes the sum of the allocated bandwidths for all connections) toward B_T, the objective (5b) is clearly being pursued. When $\sum_{i=1}^{N}\sum_{j=1}^{M}(B_i^j/B_T)V_{ij}$ exceeds 1 (i.e., $\sum_{i=1}^{N}\sum_{j=1}^{M}B_i^jV_{ij} > B_T$), the η^ζ factor plays a role in changing the current evolutionary direction of the HNN by imposing a high penalty on the violation of the capacity constraint (5c). Hence, the μ_2 term drives the system toward higher resource utilization (5b) while satisfying the capacity constraint [inequality (5c)] at the same time.

On the other hand, the absolute value function is noteworthy. The linear function (rather than a quadratic function) helps solve the optimization problem by just changing the input bias current of the HNN, because that is the actual data provided by the user to the neural network (see Section III-C). Therefore, the above two terms (i.e., μ_1 and μ_2) completely characterize the dynamic CAC problem.

The μ_3 term prevents unusable QoS levels from being included in the discovered QoS vector, when the system serves different grades of service (e.g., premium or economy services). The remaining terms (μ_4 and μ_5) are auxiliary factors that ensure rapid convergence to correct, stable states of neurons. The μ_4 term nudges the state of each neuron toward convergence to one of the $2^{N\times M}$ corners of the $(N\times M)$-dimensional hypercube, defined by $V_{ij}\in\{0,1\}$. Finally, the μ_5 term reflects a physical restriction that one connection must be allocated by one QoS level so that only one neuron will be ON in each row.

Note that the HNN evolving with the proposed energy function in (10) always converges to its stable state whenever the function is mapped onto the two-dimensional (2-D) energy function of (3). Takefuji [29] has already proved convergence of gradient descent with energy function in (3). Because all the elements of (10) correspond to $\mathbf{T}$ and $\mathbf{I}$ terms in (3), the proposed energy function maintains convergence of the HNN (Section III-C).

C. Interconnection Weights and Bias Currents

HNN evolves along a trajectory over which an energy function decreases monotonically. Rewriting (2) and (4) in terms of a 2-D HNN leads to

$$\frac{dU_{ij}}{dt} = -\frac{U_{ij}}{\tau} + \sum_{k=1}^{N}\sum_{l=1}^{M}T_{ij,kl}V_{kl} + I_{ij} \tag{11}$$

$$\frac{dU_{ij}}{dt} = -\frac{U_{ij}}{\tau} - \frac{\partial E}{\partial V_{ij}} \tag{12}$$

[18]–[20]. By substituting (10) into (12), one obtains

$$\frac{dU_{ij}}{dt} = -\frac{U_{ij}}{\tau} - \frac{\mu_1}{2}C_{ij} - (-1)^{1-\zeta}\cdot\frac{\eta^\zeta\mu_2}{2}\cdot\frac{B_i^j}{B_T} - \frac{\mu_3}{2}\psi_{ij} - \frac{\mu_4}{2}(1-2V_{ij}) + \mu_5\left(1-\sum_{l=1}^{M}V_{il}\right). \tag{13}$$

Equating the corresponding coefficients in (11) and (13), the interconnection weights and bias currents are found to be

$$T_{ij,kl} = \mu_4\delta_{ik}\delta_{jl} - \mu_5\delta_{ik} \tag{14}$$

$$I_{ij} = -\frac{\mu_1}{2}C_{ij} - (-1)^{1-\zeta}\cdot\frac{\eta^\zeta\mu_2}{2}\cdot\frac{B_i^j}{B_T} - \frac{\mu_3}{2}\psi_{ij} - \frac{\mu_4}{2} + \mu_5. \tag{15}$$

It is noted that a feasible solution (i.e., QoS vector) can be obtained to any type of problem by properly adjusting the input bias currents as given in (15), because all the parameters directly related to the problem (e.g., C_{ij}, B_i^j, and ψ_{ij}) are included in the input biases. Numerical solution of (12) follows from

$$U_{ij}(t+\Delta t) = U_{ij}(t) + \Delta t\left\{-U_{ij}(t) - \frac{\partial E}{\partial V_{ij}}\right\} \tag{16}$$

by (first order) Euler's technique [30]. Here, Δt is the time interval over which output voltages of neurons are observed and updated.

Finally, the output function of the neurons is given by

$$V_{ij} = f_{ij}(U_{ij}) = \frac{1}{1+e^{-\alpha_{ij}U_{ij}}}. \tag{17}$$

Stable (final) states of the neurons exhibit the assigned QoS levels (i.e., bandwidth) of connections. For example, $V_{ij}=1$ implies that connection i is assigned the QoS level j.

A major advantage of the HNN is its ability to speedily adapt to dynamic networks without any modification to any of the (interconnection) weights [see (14)], thereby rendering HNN-based CAC algorithm quite attractive for real-time applications.

D. Dynamic CAC

An important issue in CAC problems concerns a policy for controlling new and handoff connections. Of course, no new connection blocking and handoff-connection dropping is ideally desirable. Practically, it is impossible to achieve this goal. Instead, some algorithms provide probabilistic QoS guarantees by keeping handoff connection-dropping probability below a specified level while, at the same time, minimizing new connection-blocking probability [2]–[5], [9], [11]. In the dynamic approach (for adaptive multimedia services), connection-blocking and -dropping probabilities can be obviously minimized even if the offered load is relatively high, because the system will not force connections to block or drop unless there is enough bandwidth to support all the connections with their minimum QoS levels. Therefore, high resource utilization and fairness in bandwidth allocation are the most important criteria in the dynamic approach. It is noted that the proposed HNN can naturally pursue the criteria (see Section III-B).

The operation procedure of the proposed dynamic CAC algorithm is depicted in Fig. 1. At the start of every frame, the algorithm checks whether or not any new or handoff connection

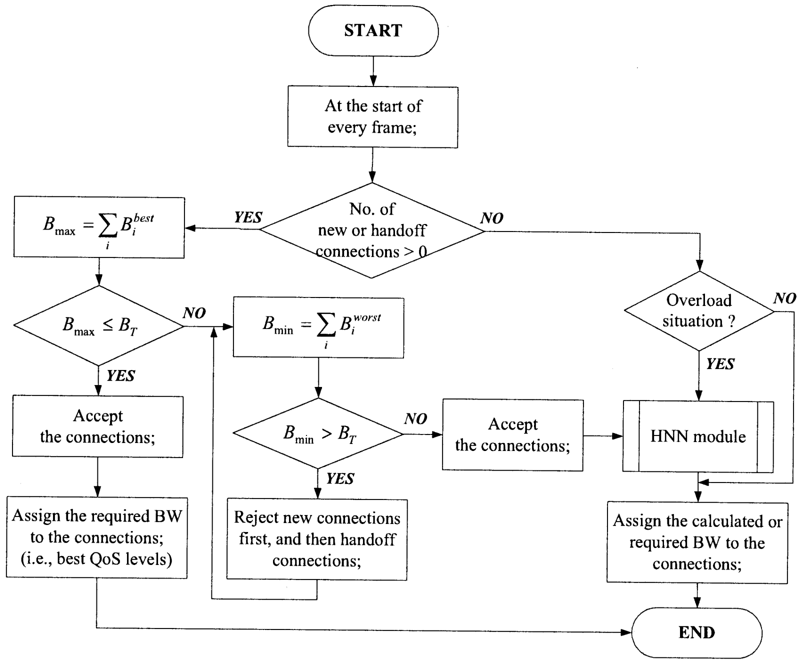


Fig. 1. Schematic diagram of the proposed dynamic CAC algorithm.

occurs. When no new or handoff connection arrives, bandwidth assignments for the existing connections (in the current frame) are just recalculated by the HNN module, if necessary. This is so because the traffic characteristics of multimedia services are usually quite dynamic (i.e., time varying) in nature.

When new or handoff connections request entry, they are all accepted at a cell if there is enough bandwidth to provide the bandwidth requirements corresponding to their best QoS levels (i.e., $\sum B_i^{\text{best}} \le B_T$). If the sum of bandwidth requirements needed to serve their worst QoS levels exceeds the bandwidth capacity (i.e., $\sum B_i^{\text{worst}} > B_T$), new connections are blocked first and then handoff connections are dropped until each of the rest of the connections can be served at minimum QoS level. This policy assigns priority to handoff connections over new connections, thereby keeping the dropping probability lower than the blocking probability. After doing this, existing connections compromise their QoS levels to make available some bandwidth for accepting new or/and handoff connections during the current frame. The QoS levels are calculated in the HNN module keeping in view high utilization of the system and fairness in resource sharing.

However, the HNN module includes an additional function that performs a local search with a greedy algorithm, since there is always a possibility that the network finds a local optimum that is located near the global optima. The greedy algorithm tries to increase or decrease the QoS levels of connections with a priority scheme that favors small changes in bandwidth. Note that it does not compromise on computation time because the number of combinations needed for finding a better (or optimal) solution is very small.

IV. Results and Discussion

Computer simulations using stochastic self-driven discrete-event models are used to evaluate the performance of the proposed algorithm. To highlight the performance characteristics, we have chosen the proportional degradation algorithm [8] as a reference. Also, we have applied identical arrival and handoff patterns to the reference in the interest of fair comparison. Our main interest in the experiments is high utilization of the system and fairness of allocated resources without any blocking and dropping connections in a given simulation environment (scenario).

A. Simulation Environment

As a representative multimedia wireless network, the wireless ATM (WATM) network has been considered for simulation. The network constructs a ring consisting of eight cells, as in [8] and [11]. The probability of a connection handing off to any neighboring cell is the same. Each cell has 3.072 Mb/s (channel bit rate) for uplink channel and one DLC-PDU (i.e., WATM packet) consists of 64 bytes, including 48 bytes payload. Average (inter-) arrival and average handoff times (i.e., average sojourn time) are assumed to be exponentially distributed. The performance of the optimization mechanism is investigated by specifically assuming an ideal wireless channel. In particular, the channel is free of noise, fading, and any other interference affecting DLC-PDU transmission. Employing a more realistic channel model (e.g., 10^{-4} bit error rate) might be better, especially for algorithms that employ online learning strategies. However, an error-free channel subjects the proposed algorithm to real test. This is explained below.

The algorithm does not have any training phase as it employs HNN, an unsupervised learning network. Moreover, the algorithm only operates with the collected information that describes usable QoS levels. The information is gathered via a control channel [which is highly reliable due to the heavy-rate (channel) coding scheme and error-recovery protocol of DLC layer even when the physical channel is not satisfactory] in the signaling

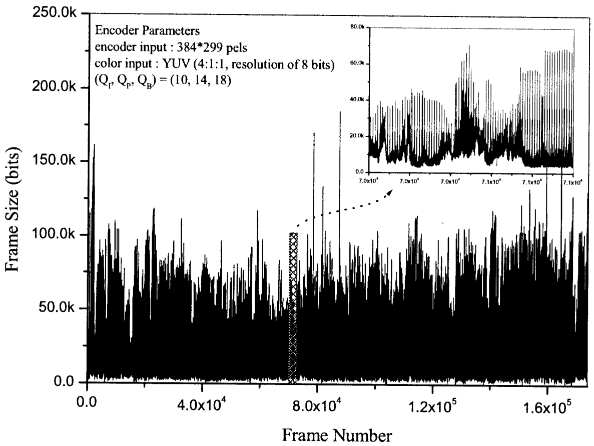
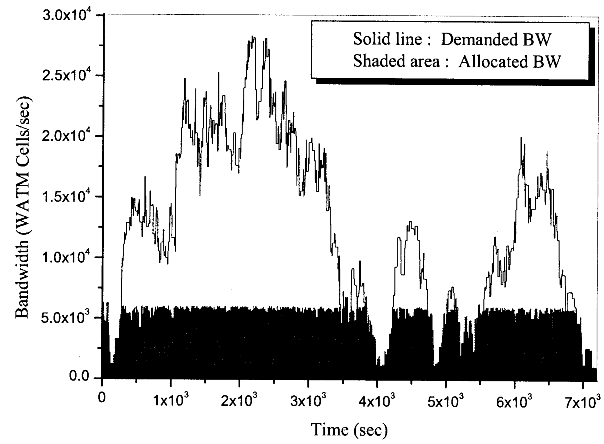
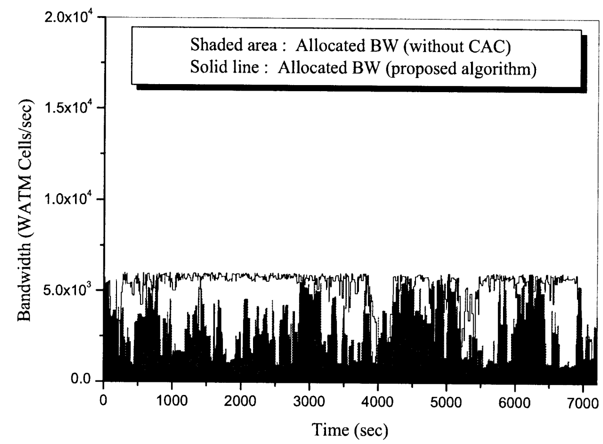


Fig. 2. Traffic characteristics of MPEG video source (*Star Wars*).



(a)



(b)

Fig. 3. Traces of various bandwidth in accordance with time. (a) Comparison of the demanded and allocated bandwidth. (b) Comparison of the allocated bandwidth with and without control.

phase. Of course, certain new or handoff connections might not be accepted under the harsh-fading channel (it is very rare, however) because they fail to exchange the signaling messages. Although such an event decreases the overall performance of the system, the mechanism of the proposed algorithm itself is not dependent upon the refusal at all. This is because it is guaranteed that all the accepted connections successfully transmit their requirements about all the feasible bandwidths to the BS. On the other hand, it is possible that it fails to renegotiate the QoS contracts of connections whose traffic rates are altered. In that case, the algorithm itself does not fail because the previous information can still be used for computing the QoS levels. This event certainly degrades the system performance, but other connection-admission algorithms are no better in this regard. Therefore, the assumption (error-free channel) is reasonable enough for testing the feasibility of the algorithm in a comparative study.

In this experiment, video, voice, and data substreams are employed as a multimedia connection. The video substream is generated with actual MPEG coded movie (*Star Wars*) captured as 384 lines by 288 pels with eight-bit color information. The sequence of MPEG I, P, and B frames used in the current experiment is IBBPBBPBBPBB$\cdots$ and there are 12 video frames in a group of picture (GOP). The video interframe rate is 41.7 ms, which is equivalent to 24 video frames per second. Fig. 2 shows the *Star Wars* MPEG source, which is characterized by mean frame size of 15.6 kb and peak frame size of 185.3 kb, yielding a burstiness factor of about 11.9. The voice and data substreams are generated according to an exponential distribution. According to the high, medium, and low QoS levels, voice traffic is generated by varying the average bit rate from 64.512 kb/s to 36.864 kb/s to 18.432 kb/s, respectively. Similarly, data traffic is generated from 36.864 kb/s to 18.432 kb/s to 9.216 kb/s, in accordance with each QoS level. The simulation is run for 7200 s (equivalent to running time of the movie). This turns out to be long enough to ensure accurate results. The details of simulation environment can be found in [14] and [15].

Evolution of the HNN consists of observing and updating the neurons' output voltages at incremental time steps Δt. Initial input voltages U_{ij} should be set to zero, but a small random noise [−0.0002,0.0002] breaks the inherent symmetry that is found when the HNN topology is symmetric [18]. Without loss of generality, the time constant τ of each neuron is set to 1; for simplicity, it is assumed that $\alpha_{ij} = \alpha$ [i.e., $f_{ij}(U_{ij}) = f(U_{ij})$]. The evolution of HNN is stopped when the output voltage of any neuron does not change by more than a threshold value ΔV_T between consecutive updates. By taking into account the criteria for determining the weighting coefficients (μ_i) (described in the Appendix) and the effect of other parameters, the following set of system parameters for simulating the HNN was decided upon:

$$\tau = 1.0; \quad \alpha = 1.0; \quad \Delta t = 10^{-4}; \quad \Delta V_T = 10^{-4};$$
$$\eta = 10; \quad \mu_1 = 1000; \quad \mu_2 = 4000; \quad \mu_3 = 8000;$$
$$\mu_4 = 800; \quad \mu_5 = 6000.$$

After reaching a stable state, each neuron is either ON (1, if $V \geq 0.5$) or OFF (0, if $V < 0.5$).

B. Simulation Results and Discussion

Figs. 3–7 plot the sample paths obtained from the system environment with average arrival time of 1 min, average departure time of 0.917 min, and average sojourn time (i.e., handoff period) of 5 min. The offered load (denoted by ρ) amounts to 0.917 because we have defined it as the ratio of *average arrival rate* to

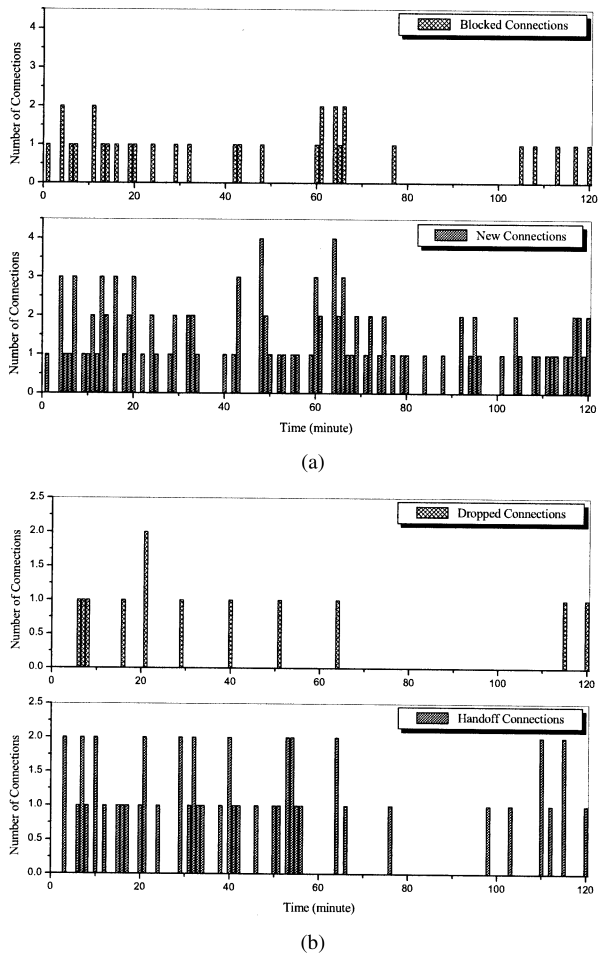


Fig. 4. Traces of the number of connections in accordance with time. (a) Number of new and blocked connections versus time. (b) Number of handoff and dropped connections versus time.

average departure rate [i.e., $\rho = (1/1.0)/(1/0.917)$]. The load is high enough for performance studies.

Fig. 3(a) compares the demanded bandwidth needed to serve all the existing connections at their best QoS levels with the allocated bandwidth after invoking the proposed algorithm. Two traces are identical whenever the demanded bandwidth is less than or equal to the available capacity. In the case of overload, the algorithm has allocated almost all the resources. It implies that the algorithm tries to maximize the utilization of shared resources. The overall utilization amounts to about 0.85, whereas the value of overload periods amounts to about 0.95. Fig. 3(b) plots the allocated bandwidths obtained with and without the proposed algorithm. The resource utilization achieved by the proposed algorithm is much higher than that of the pure scheme all through the simulation. The utilization without any CAC amounts to 0.51, whereas with the proposed algorithm it amounts to 0.85.

Fig. 4(a) shows the number of new and blocked connections when the proposed algorithm is not employed. Similarly, Fig. 4(b) presents the number of handoff connections and the number of dropped connections when the system does not use the proposed algorithm. Even if the pure scheme can serve all the admitted connections by their best QoS levels, the blocking and dropping probabilities (i.e., 0.178 and 0.101, respectively) become impractically high. In addition, the existing connections may be disconnected (but the figure is not plotted here) due to the dynamic nature of the multimedia traffic. Although the disconnection probability is not so high (i.e., 0.0018 in this simulation), any disconnection is undesirable for both service providers and users. It is noted that the new and handoff connections are not rejected at all and that the existing connections are not revoked in the simulation whenever the proposed algorithm is employed.

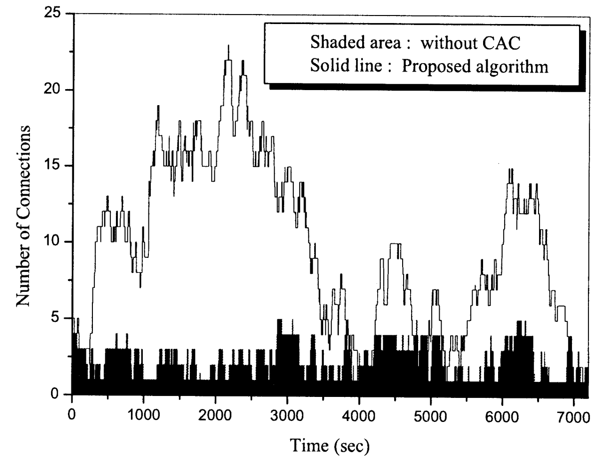


Fig. 5. Number of connections served in a cell versus time.

Fig. 5 shows the number of connections served in a cell (with and without employing the proposed algorithm). The algorithm can serve a maximum of 23 and an average of 9.58 connections simultaneously without any rejection, compared with a maximum of 5 and an average of 2.38 connections without control. This amounts to a gain by a whopping 460%. The average gain is about 400%.

Fig. 6 depicts fairness in resource sharing [defined as in (6)]. The figure shows that fairness is mostly distributed in the vicinity of the completely fair 1 without unacceptable variations. The average fairness (in overload situations) amounts to about 0.95 and its standard deviation approximately reaches 0.06. It implies that the algorithm distributes wireless resources quite fairly among all the connections.

Fig. 7 shows the QoS levels diluted to serve all the existing connections. The advantages shown in the previous figures (Fig. 6) are obtained at the expense of service quality, which is depicted in Fig. 7. The mean of QoS levels during overload periods amounts to about 11.4. The value corresponds to a moderate service quality (viz., medium quality for video service, high quality for voice service, and medium quality for data service). In other words, the algorithm gracefully degrades each service quality a little in order to provide some bandwidth for new or handoff connections (in the overload case). It implies that the proposed algorithm can maximize the statistical multiplexing gain without unduly degrading the service grade unless the offered load is extremely heavy.

From the results, it is noted that the proposed algorithm maximizes the resource utilization and the statistical multiplexing gain while fairly distributing wireless resources among different connections. Further, a moderate quality of service is also assured. Maximization of the statistical multiplexing gain and re-

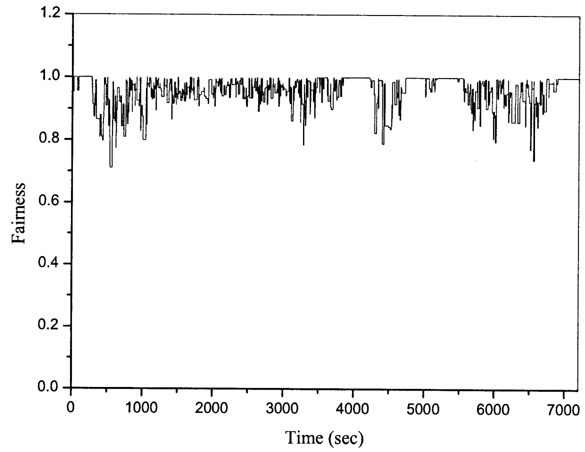


Fig. 6. Fairness in resource sharing versus time.

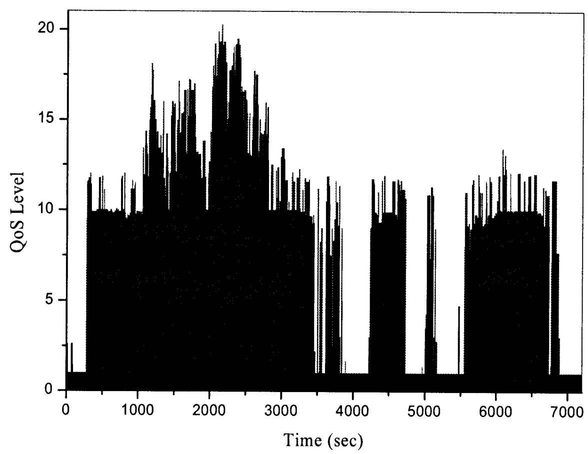


Fig. 7. QoS levels versus time.

source utilization is of interest to the network service provider, while minimization of the rejection probability and QoS level (i.e., high service quality) would be more attractive to the service subscribers in practice. The proposed algorithm provides an effective framework for simultaneously satisfying the expectations of both sides. This is because maximizing the statistical multiplexing gain and resource utilization is equivalent to minimizing the rejection probability and QoS level (as long as the scarce resource permits). Considering a more practical service scenario, the algorithm can be used for providing different grades of service (at different costs), as described in Section III-B.

At this juncture, we provide a comparative study of the results returned by the proposed algorithm and the reference (i.e., Xiao's algorithm [8]).

Figs. 8(a)–(d) show the system performance (resource utilization, fairness, number of connections, and QoS levels) when average handoff rate (1/period) varies and the offered load is 0.917 (i.e., average arrival rate of 1 and average departure rate of 1.091 [unit: connections/minute)]. Fig. 8(a) shows average resource utilization obtained with the algorithms. The proposed algorithm is superior to Xiao's algorithm with regard to the utilization performance. It amounts to approximately 10% improvement in both the overall and the overload period situations. Fig. 8(b) exhibits performance in terms of average fairness on bandwidth allocation of the algorithms. Note that the results were recorded only under overload, because fairness measure is meaningful only for distributing scarce bandwidth to a large number of existing connections. Resource allocation of the proposed algorithm seems to be fairer than that of the reference. Fairness of 0.95 is attained by the proposed algorithm while the reference only achieves fairness of 0.8. The performance improvement amounts to about 15%. From Fig. 8(a) and (b), it is seen that utilization and fairness are maintained at a constant level for the most part, without regard to the average handoff period (1/rate). The objective of maximizing resource utilization while fairly distributing the resources is thus achieved.

Fig. 8(c) depicts the number of connections served by the algorithms. The statistical multiplexing gain (i.e., connection-blocking and -dropping probabilities) can be optimized to the same extent by the proposed algorithm as well as the reference. This is because both schemes allow new and handoff connections to enter the network as long as the sum of bandwidth requirements needed to serve their worst QoS levels does not exceed the bandwidth capacity. Fig. 8(d) presents the average QoS levels degraded to serve all the existing connections. The proposed algorithm provides higher service quality to all the existing connections than does Xiao's algorithm (it accomplishes approximately 1.5 and 2.5 times higher QoS levels under overall and overload periods, respectively). As a result, the proposed algorithm can provide premium service when the reference can provide only economy level service. From Fig. 8(c) and (d), we observe that, regardless of average handoff rate, the algorithm can accept more connections simultaneously without unduly degrading service quality.

Thus, it follows that the performance of the proposed algorithm is reasonably insensitive to average handoff rate. The proposed algorithm exhibits the performance advantages over the reference in all cases.

V. CONCLUSION

This paper has proposed a dynamic CAC algorithm for multimedia wireless networks. The algorithm employs a Hopfield-type neural network. The algorithm can deal with a variety of service profiles by employing layered encoding or a data-rate-reduction scheme. By slightly degrading the QoS levels perceived by the existing connections in order to allocate some bandwidth for new or handoff connections, the algorithm can not only maximize the resource utilization, but also significantly minimize the (connection) blocking and dropping probabilities (i.e., maximizing the statistical multiplexing gain) while fairly apportioning available resources to different connections. The compromised QoS levels of existing connections and the QoS levels of new or handoff connections are computed by an HNN, thereby providing the required speed for real-time operation.

Computer simulations show significant gains in terms of minimizing the blocking and dropping probabilities and maximizing resource utilization. The algorithm maintains fairness in resource sharing while providing acceptable QoS levels (e.g.,

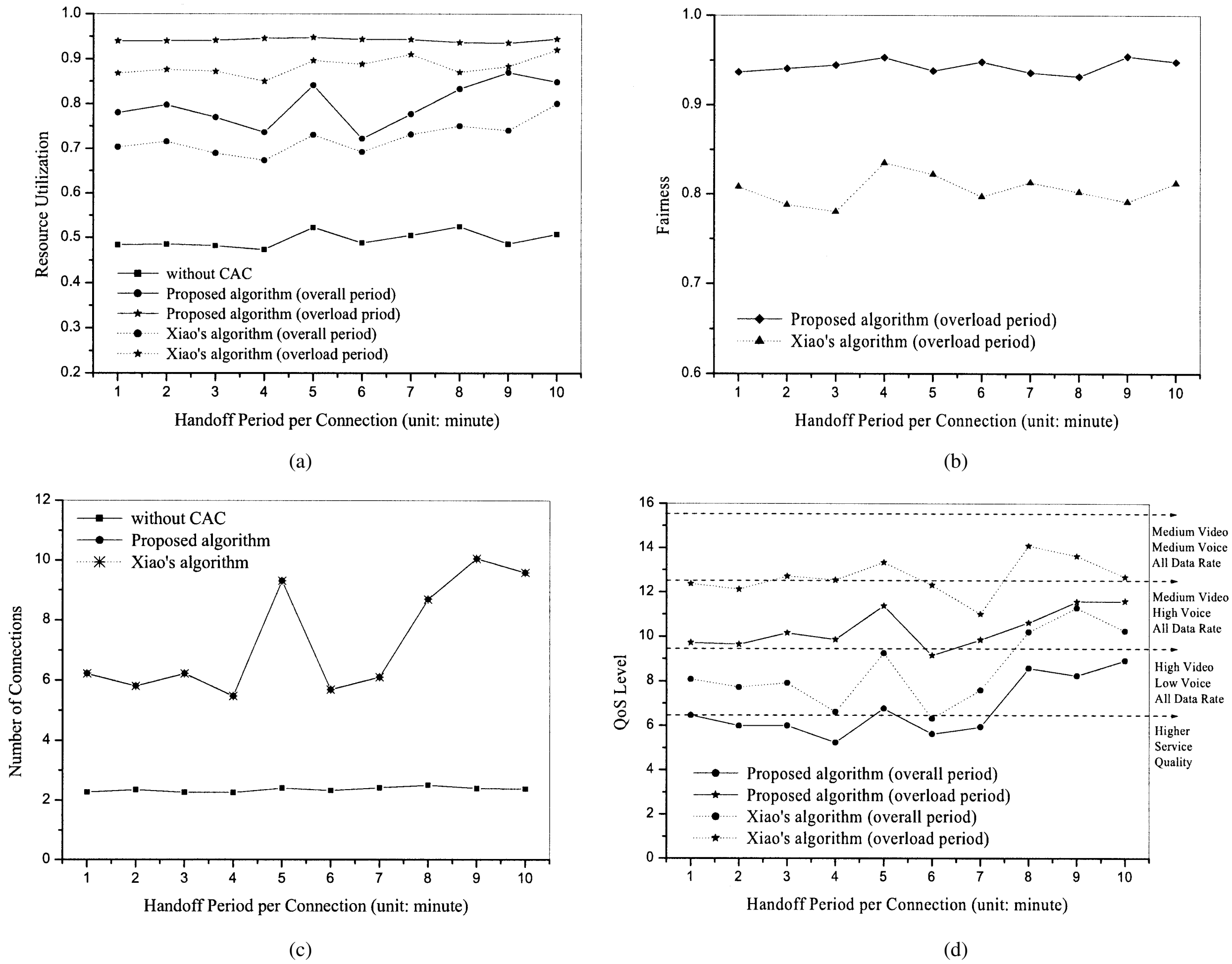


Fig. 8. Performance comparison in accordance with average handoff period (1/rate). (a) Resource utilization versus handoff period, (b) fairness versus handoff period, (c) number of connections versus handoff period, and (d) QoS level versus handoff period.

medium quality levels) to the connections. Furthermore, the performance advantages are insensitive to the handoff rate.

APPENDIX
CRITERIA FOR DETERMINING THE WEIGHTING COEFFICIENTS

In Section III-B, an energy function [described by (10)] for solving the dynamic CAC problem was proposed. For the energy function, the values of weighting coefficients (μ_i) for simulating the HNN were decided in Section IV-B, according to the criteria derived here.

In order to have minimum points with respect to output voltages (V_{ij}) of neurons, the energy function must be positive definite; it is necessary that the second derivatives be positive, i.e.,

$$\frac{\partial^2 E}{\partial V_{ij}^2} > 0. \tag{18}$$

Substituting (10) into (18), we have

$$\mu_4 < \mu_5. \tag{19}$$

Since the μ_1 term reflects the objectives [(5a) and (5b)] whereas the μ_2 term oversees both objective (5b) and constraint (5c) satisfaction, it is obvious that the μ_2 term is more important than the μ_1 term. Therefore, it is seen that

$$\mu_1 < \mu_2. \tag{20}$$

Let us investigate the initial state of the HNN. All the input voltages of neurons equal zero. This approach was considered in [18]. Input voltage-changing (increasing or decreasing) rates corresponding to the objective and constraint of the problem and the QoS indication matrix are given, respectively, by

$$\Re_1 = -\frac{\mu_1}{2}C_{ij} - (-1)^{1-\zeta}\cdot\frac{\eta^{\zeta}\mu_2}{2}\cdot\frac{B_i^j}{B_T} \tag{21}$$

$$\Re_2 = -\frac{\mu_3}{2}. \tag{22}$$

First, assume that the sum of the (current) calculated bandwidth does not exceed the available capacity (i.e., $\zeta = 0$). Equation (21) can be rewritten as

$$\Re_1^+ = -\frac{\mu_1}{2}C_{ij} + \frac{\mu_2}{2}\cdot\frac{B_i^j}{B_T} < \frac{-\mu_1+\mu_2}{2} \tag{23}$$

where an inequality is derived from $0 < C_{ij} \leq 1$ and $0 < B_{ij} < B_T$. Equation (23) indicates that the input voltage of the neuron affected by μ_1 and μ_2 terms is increasing because $\mu_1 < \mu_2$.

In order to ensure that there is no convergence toward service levels never offered by the service provider, it is necessary that $\Re_2$ (decreasing rate) should be much larger (faster) than $\Re_1^+$ (increasing rate). Referring to (20), (22), and (23), the relation between μ_2 and μ_3 can be seen to be

$$\mu_3 \gg \mu_2. \tag{24}$$

Let us now investigate the case when $\zeta = 1$. The total (current) calculated bandwidth exceeds the available capacity. Then, (21) becomes

$$\Re_1^- = -\frac{\mu_1}{2}C_{ij} - \frac{\eta\mu_2}{2} \cdot \frac{B_i^j}{B_T} < -\frac{\mu_1 + \eta\mu_2}{2}. \tag{25}$$

In order to prohibit the HNN from converging toward invalid states, producing a solution violating the constraint in inequality (5c), $\Re_1^-$ (decreasing rate) should be much larger (faster) than $\Re_2$ (decreasing rate). Referring to (20), (22), and (25), the following relation can be derived:

$$\eta\mu_2 \gg \mu_3. \tag{26}$$

It is clear that the μ_4 should be lower than the μ_1 because the μ_4 term helps the μ_1 term in driving the HNN toward speedy convergence to one of the $2^{M\times N}$ corners of the $(M \times N)$-dimensional hypercube. That is

$$\mu_1 > \mu_4. \tag{27}$$

It is also obvious that the μ_5 term is a more basic constraint than the μ_2 term because all the existing connections must be assigned at only one QoS level, no matter what the utilization and fairness are. Furthermore, the value of μ_5 must be much less than that of μ_3, because the infeasible QoS levels cannot be assigned to the connections. Hence, it follows

$$\mu_2 < \mu_5 \ll \mu_3. \tag{28}$$

Arranging (19), (20), (24), (26), (27), and (28), the criteria for determining the weighting coefficients can be seen to be

$$\mu_4 < \mu_1 < \mu_2 < \mu_5 \ll \mu 3 \ll \eta\mu_2. \tag{29}$$

Acknowledgment

The authors would like to thank B. Li and the reviewers for their thoughtful, constructive comments and suggestions.

References

[1] M. Schewartz, "Network management and control issues in multimedia wireless networks," *IEEE Pers. Commun.*, pp. 8–16, June 1995.
[2] R. Ramjee, D. Towsely, and R. Nagarajan, "On optimal call admission control in cellular networks," *Wireless Networks*, vol. 3, no. 1, pp. 29–41, 1997.
[3] C. Ho and C. Lea, "Improving call admission policies in wireless networks," *Wireless Networks*, vol. 5, no. 4, pp. 257–265, 1999.
[4] O. Lázaro and D. Girma, "A Hopfield neural-network-based dynamic channel allocation with handoff channel reservation control," *IEEE Trans. Veh. Technol.*, vol. 49, no. 5, pp. 1578–1587, 2000.
[5] B. M. Epstein and M. Schwartz, "Reservation strategies for multimedia traffic in a wireless environment," in *Proc. IEEE VTC'95*, July 1995, pp. 165–169.
[6] T. Kwon, I. Park, Y. Choi, and S. Das, "Bandwidth adaptation algorithms with multi-objectives for adaptive multimedia services in wireless/mobile networks," in *Proc. ACM Workshop on Wireless Mobile Multimedia*, Aug. 1999, pp. 51–58.
[7] Y. Xiao, P. Chen, and Y. Wang, "Optimal admission control for multi-class of wireless adaptive multimedia services," *IEICE Trans. Commun.*, vol. E84-B, no. 4, pp. 795–804, 2001.
[8] Y. Xiao and P. Chen, "Improving degradation and fairness for mobile adaptive multimedia wireless networks," in *Proc. IEEE ICCCN'01*, Oct. 2001, pp. 598–601.
[9] J. Misic, S. Chanson, and F. Lai, "Admission control for wireless networks with heterogeneous traffic using event based resource estimation," in *Proc. IEEE ICCCN'97*, Sept. 1997, pp. 262–269.
[10] S. Choi and K. G. Shin, "Predictive and adaptive bandwidth reservation for handoffs in QoS-sensitive cellular networks," in *Proc. ACM SIGCOMM'98* , Sept. 1998, pp. 155–166.
[11] B. M. Epstein and M. Schwartz, "Predictive QoS-based admission control for multiclass traffic in cellular wireless networks," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 523–534, Mar. 2000.
[12] R. G. Cheng, C. J. Chang, and L. F. Lin, "A QoS-provisioning neural fuzzy connection admission controller for multimedia high-speed networks," *IEEE/ACM Trans. Networking*, vol. 7, pp. 111–121, Feb. 1999.
[13] Q. Ren and G. Ramamurthy, "A real-time dynamic connection admission controller based on traffic modeling, measurement, and fuzzy logic control," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 184–196, Feb. 2000.
[14] M. R. Sherif, I. W. Habib, M. Nagshineh, and P. Kermani, "Adaptive allocation of resources and call admission control for wireless ATM using genetic algorithms," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 268–282, Feb. 2000.
[15] C. G. Kang, C. W. Ahn, K. H. Jang, and W. S. Kang, "Contention-fee distributed dynamic reservation MAC protocol with deterministic scheduling ($\mathrm{C-FD^3R}$ MAC) for wireless ATM networks," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 1623–1635, Sept. 2000.
[16] C. G. Kang, Y. J. Kim, and M. J. Hwang, "Implicit scheduling algorithm for dynamic slot assignment in wireless ATM networks," *Electron. Lett.*, vol. 34, no. 24, pp. 2309–2311, 1998.
[17] X. Iera, A. Moinaro, S. Marano, and F. Petulla, "A MAC protocol for wireless access to ATM networks," in *Proc. IEEE VTC'99*, Sept. 1999, pp. 1058–1062.
[18] M. K. M. Ali and F. Kamoun, "Neural networks for shortest path computation and routing in computer networks," *IEEE Trans. Neural Networks*, vol. 4, pp. 941–954, Nov. 1993.
[19] C. W. Ahn, R. S. Ramakrishna, C. G. Kang, and I. C. Choi, "Shortest path routing algorithm using Hopfield neural network," *Electron. Lett.*, vol. 37, no. 19, pp. 1176–1178, 2001.
[20] J. J. Hopfield and D. W. Tank, "Neural computations of decisions in optimization problems," *Biol. Cybern.*, vol. 52, pp. 141–152, 1985.
[21] Y. M. Huang and R. M. Chen, "Scheduling multiprocessor job with resource and timing constraints using neural networks," *IEEE Trans. Syst., Man, Cybern. B*, vol. 29, pp. 490–502, Aug. 1999.
[22] D. Girma and O. Lázaro, "Hopfield neural network DCA algorithm with combined spatial and temporal control schemes for cellular call capacity optimization," *Electron. Lett.*, vol. 37, no. 13, pp. 855–856, 2001.
[23] W. J. Li and T. Lee, "Hopfield neural networks for affine invariant matching," *IEEE Trans. Neural Networks*, vol. 12, pp. 1400–1410, Nov. 2001.
[24] J. S. Jang, C. T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Englewood Cliffs, NJ: Prentice-Hall, 1997.
[25] J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," *Proc. Nat. Acad. Sci.*, vol. 81, pp. 3088–3092, May 1984.
[26] I. Rubin and K. K. Chang, "On improving voice quality under dynamic encoding algorithms in ATM networks," in *Proc. IEEE Symp. Computers and Communications*, June 1995, pp. 9–15.
[27] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *J. Oper. Res.*, vol. 49, pp. 237–252, 1998.
[28] D. D. Luong and J. Bíró, "On the proportional fairness of TCP Vegas," in *Proc. IEEE GLOBECOM'01*, Nov. 2001, pp. 1718–1722.
[29] Y. Takefuji, *Neural Network Parallel Computer*. Boston, MA: Kluwer, 1992.
[30] K. Atkinson, *Elementary Numerical Analysis*. New York: Wiley, 1993.

Chang Wook Ahn (S'02) was born in Hae-Je, Chonnam, Korea, in 1977. He received the B.S. and M.S. degrees in electrical engineering from Korea University, Seoul, Korea, in 1998 and 2000, respectively. He is currently working toward the Ph.D. degree at the Department of Information and Communications, Kwang-Ju Institute of Science and Technology (K-JIST), Gwangju, Korea.

From January to August 2003, he was a Visiting Scholar in the Illinois Genetic Algorithms Laboratory (IlliGAL), University of Illinois at Urbana-Champaign (UIUC). His current research interests include the design and analysis of MAC protocols and QoS-oriented routings for broadband radio access networks and mobile *ad-hoc* networks. In particular, he has been working on intelligent wireless networks using neuro-fuzzy and evolutionary computing.

Mr. Ahn is a Student Member of IEEE Neural Network Society.

R. S. Ramakrishna (M'83–SM'98) received the Ph.D. degree from the Indian Institute of Technology, Kanpur, in 1979.

From 1980 to 1996, he was an Assistant Professor, Associate Professor, and Professor in the Department of Computer Science and Engineering, Indian Institute of Technology, Bombay. He is currently a Professor in the Department of Information and Communications, Kwang-Ju Institute of Science and Technology (K-JIST). His current research interests include neural networks, evolutionary algorithms, parallel and distributed computing, computer graphics, and quantum computing.