

Enhancing quality of service provisioning in wireless ad hoc networks using service vector paradigm

Didem Gozuppek¹, Symeon Papavassiliou^{2*†} and Nirwan Ansari¹

¹*Department of Electrical and Computer Engineering New Jersey Institute of Technology Newark, NJ, 07102 U.S.A.*

²*Electrical and Computer Engineering Department Network Management & Optimal Design Lab (NETMODE) National Technical University of Athens (NTUA) 9 Iroon Polytechniou str. Zografou, 15780, Athens, Greece*

Summary

Emerging real-time communications and multimedia applications necessitate the provisioning of Quality of Service (QoS) in Internet. Recently, a new concept, referred to as *service vector*, has been introduced to enhance the end-to-end QoS granularity, and at the same time, maintain the simplicity and scalability feature of the current differentiated services (DiffServ) networks. This work extends this concept to wireless ad hoc networks and proposes a cross-layer architecture based on the combination of delay-bounded wireless link level scheduling and the network layer service vector concept, resulting in significant power savings and finer end-to-end QoS granularity. The impact of various traffic arrival distributions and flows with different QoS requirements on the performance of this cross-layer architecture is also investigated and evaluated. Copyright © 2006 John Wiley & Sons, Ltd.

KEY WORDS: wireless ad hoc networks; quality of service; power efficiency; differentiated services

1. Introduction

The proliferation of Internet applications and services has fostered growing research activities in provisioning Quality of Service (QoS). Evolution of the Internet infrastructure from a best-effort service model to the one in which service differentiation can be provided for different users and applications, and progression of the networking environment towards the wireless domain, call for techniques that can facilitate

differentiated services in wireless networks. Moreover, wireless ad hoc networks introduce additional challenges since they have no fixed network infrastructure or administrative support [1,2].

Currently, two service models have been proposed for end-to-end QoS provisioning in the Internet: Intserv [3] and Diffserv [4]. The former provides per-flow-based resource reservation and allocation, whereas the latter aggregates individual flows and provides only a number of services to the aggregated

*Correspondence to: Prof. Symeon Papavassiliou, Electrical and Computer Engineering Department Network Management & Optimal Design Lab (NETMODE) National Technical University of Athens (NTUA) 9 Iroon Polytechniou str. Zografou, 15780, Athens, Greece.

†E-mail: papavass@mail.ntua.gr

data flows. Intserv suffers from the scalability problem, while Diffserv can only provide coarse QoS granularity. Recently, a novel distributed end-to-end QoS provisioning architecture has been proposed [5,6], which can enhance network resource utilization and end-to-end QoS granularity, while maintaining the simplicity and scalability feature of the Diffserv network architecture. Moreover, a new concept, *service vector*, was introduced by which an end host can choose different services at different routers along its data path [6].

The power limited and time-varying nature of the wireless domain makes most of the QoS provisioning techniques originally designed for the wired environment unsuitable for wireless networks. QoS provisioning in multi-hop wireless ad hoc networks has not been addressed successfully, since these networks pose further challenges owing to their infrastructure-less and multi-hop nature.

Multi-hop wireless networks gain increasing popularity, as multi-hop connections inevitably become necessary to maintain high degree of network connectivity and achieve higher data rates for large distances. Guaranteeing end-to-end delay in multi-hop wireless networks becomes an important issue due to the delay sensitive nature of the emerging real-time communications. The previously mentioned Intserv and Diffserv models attempt to address the end-to-end delay assurance problem, and the solutions hitherto proposed are mainly centered on these frameworks. For instance, authors in Reference [7] proposed a method for flow allocation along links while minimizing the total average power consumption, for end-to-end delay constrained traffic in multi-hop wireless networks. Unlike Diffserv scheme, Intserv provides end-to-end resource (bandwidth) reservation. However, since bandwidth is a scarce resource in a wireless setting, Diffserv-based solutions are considered to be more viable for end-to-end QoS assurances in multi-hop wireless networks. The introduction of Per domain behaviors (PDBs) in the Diffserv framework [8] aimed at addressing this end-to-end QoS guarantee issue. On the other hand, in Reference [9] a Diffserv-based end-to-end delay assurance mechanism for multi-hop WLANs has been proposed and is referred to as neighborhood proportional delay differentiation (NPDD). This solution is based on the proportional delay differentiation (PDD) model [10], which was originally designed for wireline networks. In NPDD model, an application chooses a certain service class via the implementation of dynamic class selection (DCS) algorithm [11]. In

this model, unlike the service vector scheme and paradigm that is adopted in our paper, each node along the path utilizes the same service class.

In our work, the proposed service vector paradigm [5,6] is extended to wireless ad hoc networks. A cross-layer architecture that combines the link level scheduling and network level service vector concept is proposed and evaluated. It is demonstrated that within this distributed architecture, the service vector paradigm can enhance the end-to-end QoS in wireless ad hoc networks by reducing the power consumption as well as providing finer QoS granularity.

The remaining of the paper is organized as follows. In Section 2, some relevant background information with respect to the service vector QoS provisioning paradigm is provided, while in Section 3, the corresponding problem is rigorously formulated within the multi-hop wireless ad hoc network framework. In Section 4, the power efficient delay bounded multi-user scheduling, which is an integral part of the overall proposed cross-layer approach, is described, while Section 5 presents the performance evaluation of the proposed methodology under different traffic scenarios. Finally, concluding remarks are given in Section 6.

2. Background Information on Service Vector Paradigm

Explicit Endpoint Admission Control with Service Vector (EEAC-SV) scheme [5,6] has been recently proposed in the literature as an enhanced QoS provisioning mechanism. In essence, this mechanism relies on the idea of explicitly providing information about the performance of each service class at each router during the probing phase, and enabling the flow to choose different service class at each router along the path during the data transfer phase. In other words, a service vector is selected at the end of the probing phase according to the QoS related information fed back by the network in the probing packets; this service vector is attached to the data packets in the data transfer phase. Assume that there are m routers along the path of a flow and n service classes are provisioned at each router. The set of n service classes can be denoted as $S = (S_0, S_1, \dots, S_{n-1})$. A flow may choose service s_i at router i , where $s_i \in S$, which may be different from service s_j it chooses at router j . A service vector can be represented by $s = (s_0, s_1, \dots, s_{m-1})$, where each element corresponds to the service class chosen at the corresponding

router along the path. The end host basically implements an optimization procedure in determining the appropriate service vector [5,6]. The optimization process aims at maximizing the benefit of the end host in utilizing the network services, subject to the constraints on the end-to-end QoS requirements of the data flow. The benefit of the end user is expressed in terms of the utility function which represents the user's level of satisfaction with the perceived QoS and characterizes how sensitive users are to the changes in QoS. To achieve a certain utility, the user pays for a cost represented by a cost function, which is determined by the corresponding pricing policy and the service vector the flow chooses. Therefore, the objective of the optimization process for the end user is to select the appropriate network services (i.e., service vector) that results in maximizing the user's benefits, based on the utility function and the associated user cost function [5,6]. It should be noted here that in this paper, the emphasis of our work is not placed on the actual optimization of the service vector selection; we rather study and evaluate the power savings and finer end-to-end QoS granularity that can be achieved in wireless ad hoc networks within the framework of the service vector paradigm.

Various end-to-end service provisioning methodologies, namely the static service mapping and dynamic service mapping schemes have been proposed for Diffserv networks. Therefore, within the context of the service vector concept, the various end to end QoS provisioning mechanisms can be categorized as follows:

Scheme 1—Conventional service (EEAC-CS) scheme (static service mapping): Currently, some Internet equipment vendors statically map the users' QoS requirements to a certain service class [12] and hence, the service vector s is a constant vector. If the measured QoS performance at the destination meets the QoS requirements, the flow is admitted. Otherwise, it is rejected. The end host only checks whether the statically mapped service class satisfies the requirements or not, and therefore, the resultant QoS granularity is $O(1)$.

Scheme 2—EEAC with single class of service (EEAC-SCS) scheme (dynamic service mapping): The service vector is a constant vector as in EEAC-CS; that is, only one service class is used along the path. However, the flow is now dynamically mapped to the available best service class. An optimization procedure that tries to find the service vector satisfying the QoS constraint and maximizing

the revenue among all the possible n service vectors is applied, and hence, the resultant QoS granularity is $O(n)$.

Scheme 3—EEAC with combination of service classes (EEAC-CSC) scheme (combination of service classes via the service vector): In this case, different service classes can be selected at the routers along the path; therefore, there are n^m possible solutions. A user side optimization model that operates on these n^m possible service vectors in order to identify the optimal one is applied [5,6], and hence, the resultant QoS granularity is $O(n^m)$.

3. Problem Formulation

In this paper, we consider a multi-hop wireless ad hoc network, where every node may play the role of routing by relaying packets towards their final destination. Therefore, in the following, we use the terms nodes and routers interchangeably. Assume that a flow going from its source to the destination passes through m intermediate routers, where the set of available service classes at each router is $S = (S_0, S_1, \dots, S_{n-1})$. After the probing phase is executed, the end host determines the service vector as $s = (s_0, s_1, \dots, s_{m-1})$, where the service class chosen at router i is denoted by $s_i \in (S_0, S_1, \dots, S_{n-1})$. The QoS parameter considered here in determining the service vector is the average end-to-end delay; that is, each service class s_i corresponds to a predetermined average delay bound $delay(s_i)$. The data transmission phase takes place in a time-slotted manner, and the average end-to-end delay bound of a data flow is inelastic; that is, the application does not care if better than required QoS is provided. This means that the user's level of satisfaction with the perceived QoS is the same as long as the provided QoS performance is within the required bounds.

To minimize the overall transmission power along the route once the service vector has been determined, we need to consider the following problem:

$$\begin{aligned} & \min E\{\bar{P}\} \\ \text{s.t. } & E\{D_i\} \leq \text{delay}(s_i) \forall i \in (0, 1, \dots, m-1) \end{aligned}$$

where $\bar{P} = \lim_{n \rightarrow \infty} \sum_{i=0}^{m-1} P_{i,n}$; $P_{i,n}$ is the power in time slot n at router i , D_i is the delay experienced at router i , and s_i is the service class chosen at router i .

Apparently, the above problem can be transformed to the link level scheduling problem of minimizing the

average transmit power subject to the average delay constraints of all the service class buffers. Delaying communication by decreasing the transmission rate to save power is commonly used in wireless systems [13,14].

The major merit of the service vector scheme (*EEAC-CSC*) is that it allows a flow to choose a service class with less stringent delay guarantees in some part of the network, even though that service class might be unavailable in some other part of the network. This way, the transmission rate can be decreased at that node, which in turn reduces the power consumption. For instance, suppose that the service classes 0, 1, and 2 correspond to average delay bounds of 100 ms, 200 ms, and 300 ms, respectively. Furthermore, assume that there are three nodes along the data path, and the average end-to-end delay bound of the data flow is 750 ms. *EEAC-CS* scheme results in the usage of Class 0 along the entire path, and hence 300 ms average end-to-end delay, whereas *EEAC-SCS* scheme results in the usage of Class 1 along the data path and hence 600 ms average end-to-end delay. On the other hand, *EEAC-CSC* scheme results in the usage of Class 1, Class 1, and Class 2 along the path and consequently an average end-to-end delay of 700 ms (which still meets the end-to-end delay bound of the data flow under consideration). Since larger delay corresponds to decreased transmission rate and hence less power consumption via the implementation of an appropriate scheduling discipline, *EEAC-CSC* scheme results in the least power consumption as compared to *EEAC-CS* and *EEAC-SCS* schemes. Therefore, this cross-layer approach of using the service vector scheme in combination with an appropriate scheduling discipline can enable the network to have significant power savings, which is vital for the efficient operation of wireless ad hoc networks.

On the other hand, consider another data flow with an average end-to-end delay bound of 850 ms. *EEAC-CS* and *EEAC-SCS* schemes still result in the usage of Class 0 and Class 1, respectively, along the entire data path, whereas *EEAC-CSC* scheme results in the usage of Class 1, Class 2, and Class 2, which corresponds to an average end-to-end delay of 800 ms. In other words, *EEAC-CSC* scheme can differentiate between the two data flows having 750 ms and 850 ms average end-to-end delay bounds, whereas the other two schemes fail to achieve this differentiation. Therefore, while *EEAC-CS* and *EEAC-SCS* schemes result in the same average end-to-end power consumption for these two different data flows, *EEAC-CSC* scheme

leads to less power consumption for the data flow having 850 ms average end-to-end delay bound. Therefore, in addition to power savings, *EEAC-CSC* scheme can also enable finer QoS granularity in terms of average end-to-end power consumption.

4. Power Efficient Delay Bounded Scheduling

A power efficient delay bounded multi-user scheduler is a key element in our proposed design, where we use the term *scheduling* to refer to the decision about the transmission rate and power, based on the instantaneous buffer states. Optimal and suboptimal multi-user schedulers were proposed in References [15] and [16], where a dynamic programming technique called value iteration algorithm (VIA) is utilized in finding the optimum scheduler for both single-user and multi-user cases. A suboptimum scheduler, called log-linear scheduler, was proposed for the single-user case [15]. Furthermore, a two-stage solution was proposed as a suboptimal multi-user scheduler for the time division multiple access (TDMA) scheme [16], where the flow choice is made in the first step and the number of packets to be transmitted in a certain time slot is determined in the second step, by utilizing the optimum single-user scheduler proposed in Reference [15]. Since the high number of possible states in the VIA increases considerably, the computational complexity of the optimum multi-user scheduler, decoupling the solution into these two separate stages helps to eliminate this problem.

In our work, the proposed suboptimal TDMA scheduler is modified by having the scheduler to transmit according to the suboptimal log-linear scheduler rather than the optimal single flow scheduler in the second step of the algorithm. There are basically three reasons for this design choice. First of all, the number of possible states in VIA grows exponentially as the buffer size and the number of queues in the system increase. In our implementation, three service classes: namely expedited forwarding (EF), assured forwarding (AF), and best effort (BE), are provisioned at each router. Therefore, the three buffers corresponding to these service classes do not contribute significantly to the computational complexity of the VIA. However, finding the optimal scheduler becomes computationally intensive as the buffer sizes increase. Second, the Lagrangian value ε , which is a parameter of the cost function in VIA, was found to be mathematically intractable, even for the single user

optimum scheduler. Third, VIA requires detailed knowledge about the arrival distribution. In situations where the optimal scheduler is adapted over time as the arrival distribution changes the implementation of VIA may lead to an intractable design. In principle, there are two means to obtain information about the arrival distribution. The first option is that the arrival distributions can be measured in real time; however, this option is not feasible not only because it introduces additional implementation complexity, but also because the measurement results might be inaccurate and can lead to erroneous scheduler decisions. The second option is that each router can compute its output distribution and send this information to the next router along the path. Nevertheless, this option also introduces extra messaging overhead in the network, while the computation of the output distribution is not trivial. This can be clearly illustrated by the one buffer case below, where the output distribution at a router is given as follows:

$$\text{Prob}(b_n = j) = \sum_{k=0}^L s_k \times \alpha_{j,k}$$

where $s_k = \text{Prob}(x_n = k)$ and
 $\alpha_{j,k} = \text{Prob}(u_n = j/x_n = k)$

Here, b_n denotes the number of packets transmitted by the router at time slot n , x_n is the number of packets in the buffer at the beginning of time slot n , u_n is the number of packets chosen for transmission at the beginning of time slot n , L is the buffer size, and $\alpha_{j,k}$ denotes the corresponding scheduling actions. Since each router knows its own scheduler actions, $\alpha_{j,k}$ values are known; therefore, the router only needs to compute its vector of stationary probabilities of the buffer states, $s = [s_0 \ s_1 \ \dots \ s_L]$. The stationary probability of buffer state y can be expressed as $s_y = \sum_{i=0}^L \sum_{t=y-i}^y (s_i \times \text{Pr}(a_n = t) \times \alpha_{t+i-y,i})$, where a_n denotes the number of packets arrived in time slot n . Together with the fact that all the stationary buffer probabilities sum up to 1, hence $\sum_{y=0}^L s_y = 1$, a system of $L + 2$ linear equations needs to be solved.

In the general case where K buffers are available, the number of queued packets at the beginning of the n th time slot in buffer i is denoted by $x_{i,n}$ and each buffer state is represented by a $1 \times K$ vector as $[x_{1,n}, \dots, x_{K,n}]$. Buffer i has size of L_i packets and receives $a_{i,n}$ packets in the n th time slot. The number of packets scheduled for transmission is also represented by a $1 \times K$ vector as $[u_{1,n}, \dots, u_{K,n}]$, where $u_{i,n}$ denotes the number of packets chosen for

transmission from buffer i at the beginning of the n th time slot. For instance, for the two buffer case, the expression for the stationary buffer probabilities is even more complicated, than the corresponding ones for the one buffer case. The stationary probability of buffer state $[y, z]$ can be conveyed as:

$$s_{y,z} = \sum_{i=0}^y \sum_{j=0}^z \sum_{k=0}^{L_1} \sum_{m=0}^{L_2} s_{k,m} \times \alpha_{[k-i, m-j], [k, m]} \\ \times \text{Pr}(a_{1,n} = y - i) \times \text{Pr}(a_{2,n} = z - j)$$

Therefore, considering the fact that in general, several service classes may be available (for instance to support three service classes such as EF, AF, and BE traffic, three buffers are required), the computation of the arrival distribution is apparently intensive.

Moreover, with reference to *Scheme 3 (EEAC-CSC)*, rather than being computationally involved, it is mathematically impossible to compute the arrival distribution for a certain buffer, due to the fact that when a certain number of packets leave the buffer of a given service class at a router, does not necessarily mean that they are going to use the same service class at the next router, since *Scheme 3 (EEAC-CSC)* permits a data flow to use different service classes at different routers along the path. Therefore, even the suboptimal multi-user scheduler where the optimal single user scheduler is used in the second step, cannot be implemented for the service vector scheme. Because of these reasons, the suboptimal TDMA scheduler originally proposed in Reference [16], is modified here by using the *log-linear* scheduler in the second step of the algorithm.

Furthermore, the three conditions for zero-outage are ensured for the scheduler. First, no packet dropping is allowed. Second, reliable communication is guaranteed by having the scheduler to choose the power level P_n in time slot n such that the number of packets transmitted u_n is equal to the Shannon capacity function for a Gaussian channel [15]. Third, zero buffer overflow is ensured by guaranteeing that $x_k \geq (L_k - M_k)$, for at most one $k = 1, 2, \dots, K$, where x_k is the number of packets in buffer k , L_k is the size of buffer k , and M_k is the maximum number of packets that can arrive at buffer k in a time slot. The reason for this last condition is that only one user can transmit in a certain time slot in TDMA system. For instance if $K = 2$, $L_1 = L_2 = L$, and $M_1 = M_2 = M$, when both buffers have at least $L - 2M + 1$ packets, the scheduler transmits M packets from one of them. This way, any possibility of

buffer overflow in the next time slot is avoided. If at most one buffer has $L - 2M + 1$ packets, then the scheduler implements its scheduling action, as described below. On the other hand, the upper limit on the output rate of the scheduler is set to be equal to the maximum number of packets that can arrive at a router in a certain time slot; that is, the maximum number of packets that the scheduler can transmit in a certain time slot is equal to $\sum_{k=1}^K M_k$. The reason for this is that in order for queuing to occur at a node, outgoing packet transmission rate from that node has to be less than or equal to the packet arrival rate to that node.

Therefore, based on the above discussion, the following scheduler has been implemented:

Step 1. Flow choice: Index k of the flow chosen to transmit:

$$k = \begin{cases} l & \text{if } x_l > L_l - M_l \\ \arg \max_l \frac{x_l}{\lambda_l D_{l,0}} & \text{else} \end{cases}$$

Step 2. Number of packets:

$$u_n = \min(x_n, \lfloor \log(\kappa x_n) \rfloor)$$

where x_l denotes the number of packets at buffer l at the beginning of the time slot under consideration, L_l denotes the size of buffer l , M_l denotes the maximum number of packets that can arrive at buffer l , λ_l represents the average arrival rate to buffer l , and $D_{l,0}$ corresponds to the average delay bound of buffer l . Furthermore, as explained before, u_n denotes the number of packets chosen for transmission from the selected buffer at the beginning of time slot n , x_n denotes the number of packets at the selected buffer at the beginning of time slot n , while κ is a parameter that is chosen so that the average delay bound is satisfied. It should be noted that with reference to Step 1 of the algorithm, the first condition ensures zero buffer overflow, whereas the second condition results in choosing the flow that is closest to violating its delay bound.

5. Performance Evaluation

The performance of the proposed cross-layered architecture and corresponding methodology is achieved via modeling and simulation, using the Optimized Network Engineering Tool (OPNET). Specifically, two different types of scenarios have been considered, in both of which the route of a flow is assumed to be

predetermined. The first scenario refers to the use of a single flow from the source to the sink, where the wireless links are assumed to be AWGN channels, and allows for an in-depth evaluation of the achievable performance and corresponding trade-offs of the proposed approach. In the second scenario, two flows with different QoS requirements are considered, in order to better demonstrate the service differentiation capabilities of our proposed mechanism. In order to better evaluate and demonstrate the corresponding benefits that can be obtained, the performance results of three different types of QoS provisioning schemes, that is, Scheme 1 (EEAC-CS), Scheme 2 (EEAC-SCS), and Scheme 3 (EEAC-CSC), are presented and compared.

5.1. Models and Assumptions

Three different service classes, namely EF, AF, and BE, are considered, and the TDMA system is used as the multiple access scheme for these service classes. The time slot length is assumed to be fixed: $T_s = 0.05$ s, and for all routers along the path the buffer size of service class k is $L_k = 170$, $\forall k = 1, 2, 3$, and the maximum number of packets that can arrive at the class k buffer is $M_k = 6$, $\forall k = 1, 2, 3$. In order to study the impact of the traffic arrival distribution, either case is considered under both uniformly distributed and *On-Off* arrival distributions.

The network topology under consideration in this study is shown in Figure 1. The source node sends a probing request to the network and gathers information about each service class at the routers. The information that the routers attach to the probing acknowledgement packets refer to the availability of the various service classes at each router. The availability of each service class is determined based on the packet arrival rate to that service class buffer. The end host then determines the best service vector among the available ones. The average delay bounds for the service classes considered in this study are defined in Table I.

With reference to Figure 1, the direction of the data flow whose performance was evaluated is from node A to node E. Cross traffic is assumed to be uniformly distributed, and the maximum number of packets that can arrive in a time slot for the background traffic flows is summarized in the following Table II. Packets arrived at the router are placed in one of the three buffers depending on their service vectors; that is, the service class they are using at that router, in a FIFO manner. At the beginning of each time slot, based on

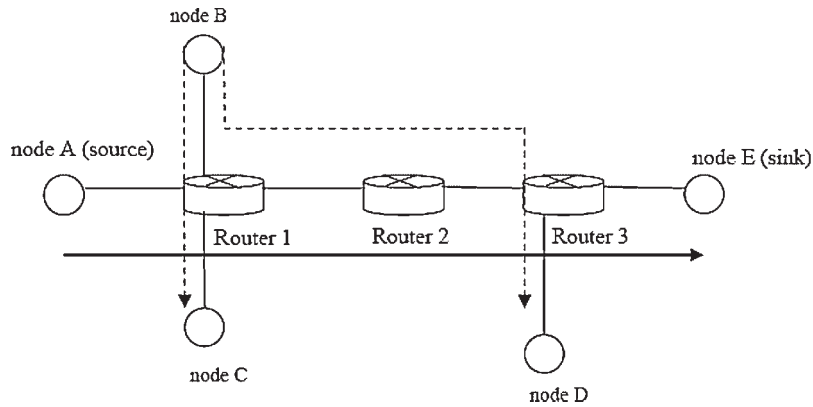


Fig. 1. The simulated network topology.

the methodology described in Section 4, the scheduler determines which buffer to serve and how many packets to transmit from that buffer at that time slot. Since TDMA is used, only one buffer is allowed to transmit its packets in a certain time slot. However, in a time slot, the selected flow is allowed to transmit multiple packets. The upper limit of the number of packets of the selected flow that the scheduler can transmit in a certain time slot is the summation of the maximum number of packets that can arrive at all the buffers of a router in a certain time slot, as explained in Section 4.

The reason for checking the availability of the service classes at each router during the probing phase is demonstrated in Figures 2 and 3, which illustrate the average delay and power consumption of the various service classes at router 1, before the flow from node A to node E starts sending traffic. As observed from Figure 3, class 2 traffic at router 1 has significantly higher power consumption than the other ones, although it is the service class with the least stringent delay bound requirement. Besides, as observed from Figure 2, the average packet delay of this service class is much smaller than its required value. This happens because the cross traffic overloads class 2 at router 1. As the arrival rate to a certain buffer increases quite considerably, the scheduler greatly increases the rate of transmission from that

buffer, mainly in order to prevent buffer overflows as well as to meet the delay bound requirement. As a result, the actual average delay of that service class becomes much smaller than its required value at the expense of enormous power consumption. Furthermore, the scheduler can guarantee zero buffer overflow provided that the maximum number of packet arrivals per time slot to each buffer is less than or equal to its upper bound. For instance, in our model, the maximum number of packets that can arrive at the class k buffer is $M_k = 6, \forall k = 1, 2, 3$. Considering that the arrival traffic is uniformly distributed, the arrival rate should not exceed three packets per time slot. For these two reasons, it is crucial to determine whether the current maximum number of packets arriving at the scheduler in a time slot is already close to its upper limit or not, so that a new flow admitted to the network should not utilize an overloaded service class buffer.

Power consumption is related to the output rate of the scheduler, which is in turn directly related to the arrival rate. Therefore, the estimated value of the packet arrival rate accurately reflects the fact that a certain service class is overloaded and can differentiate itself from the buffers that are not overloaded along the path. Consequently, estimation of the packet arrival rate is used in this work to determine the availability of a certain service class in the probing

Table I. Service class definitions.

Service class	Average delay bound
EF (Class 0)	100 ms
AF (Class 1)	150 ms
BE (Class 2)	350 ms

Table II. Summary of background traffic.

Source	Destination	Class 0 (EF) (packets/slot)	Class 1 (AF) (packets/slot)	Class 2 (BE) (packets/slot)
Node B	Node D	2	2	2
Node B	Node C	0	0	4

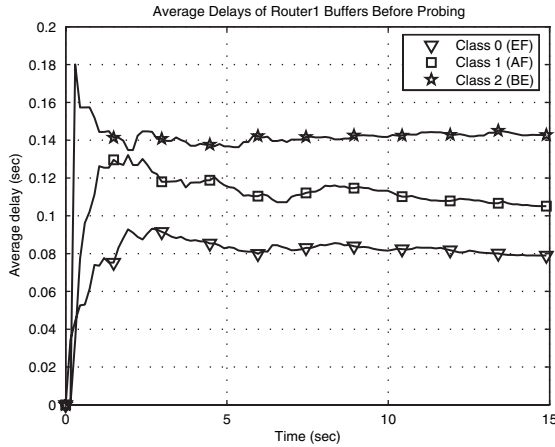


Fig. 2. Average delays at router 1 for different service classes.

phase. Exponential moving average filter is used in the estimation process [17], which is measured in terms of packets/time slot as follows:

$$\bar{\tau}_{S_j}(t) = \left(1 - e^{-\tau_{S_j}(t)/K}\right) \frac{T_s}{\tau_{S_j}(t)} + e^{-\tau_{S_j}(t)/K} \bar{\tau}_{S_j,old}(t)$$

where T_s is the time slot length in seconds, $\bar{\tau}_{S_j}(t)$ is the estimated arrival rate for service class S_j at time t , $\bar{\tau}_{S_j,old}(t)$ is the most recently updated arrival rate before t , $\tau_{S_j}(t)$ is the interval between the arrival of the previous received packet of service class S_j and the current time t , and K is a constant. At each router, $\bar{\tau}_{S_j}$ is updated when a data packet of service class S_j is received. In the probing phase, if $\bar{\tau}_{S_j} > \frac{M_{S_j}-1}{2}$, S_j is

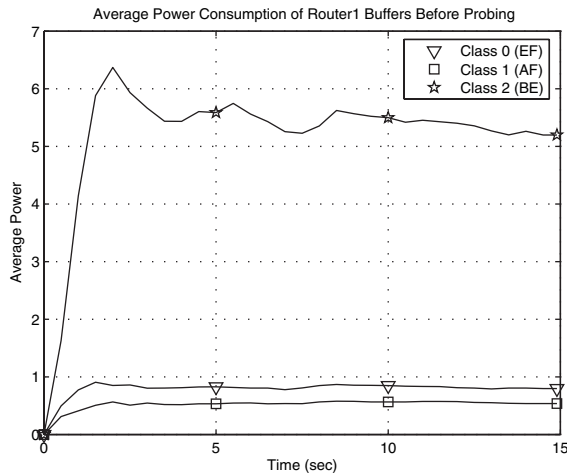


Fig. 3. Average power consumption at router 1 for different service classes.

marked as unavailable in the probe acknowledgement packet; otherwise, it is marked as available.

Selection of the constant K affects the performance of the arrival rate estimation. More specifically, a small value of K enables the estimation process to track the variation in traffic appropriately; nevertheless it cannot filter out the transient changes in the data rate. On the other hand, a large value of K can filter out these changes and hence, provide stable network performance, however, it cannot respond to the changes in the traffic arrival pattern quickly. The exponential moving average filter has the following unit sample response function:

$$h(a) = \left(1 - e^{-\tau_{S_j}^{min}/K}\right) \left(e^{-\tau_{S_j}^{min}/K}\right)^a U(a)$$

where a is the number of packet arrivals that determines the convergence time required for the measurement result to converge to the actual arrival rate, and $\tau_{S_j}^{min}$ is the minimum time between consequent packet arrivals. Since the time slot length is $T_s = 0.05$ s and the maximum number of packets that can arrive at a service class buffer is $M_{S_j} = 6, \forall j = 1, 2, 3$, $\tau_{S_j}^{min} = 0.00833$ s.

Assume that a new flow will use service class S_j with probability p_{S_j} . In order to avoid buffer overflow due to slow convergence of the exponential moving average filter, the average convergence time should be less than $\sum_j p_{S_j} L_{S_j}$ of the packet arrivals. Since $L_{S_j} = 170, \forall j = 1, 2, 3$, the average convergence time should be less than 170 packet arrivals; that is, $a < 170$. Let a_{stop} represent the convergence time where $h(a_{stop}) = -10$ db, due to the reason that $h(a)$ will have little impact on the exponential moving average result when $a > a_{stop}$. Therefore; $a_{stop} = 170$ and hence, $K \leq 12.3$. On the other hand, the smallest possible value of K stands for the case where the exponential moving average filter immediately converges to the actual measurement result; that is, $a_{stop} = 1$ and hence, $K \geq 0.0724$. Consequently, K should be in the range of $0.0724 \leq K \leq 12.3$. In this study, K was selected to be 0.35, which was found to be able to provide an accurate estimate of the actual arrival rate, while at the same time being within the above mentioned required bounds.

5.2. Numerical Results and Discussions

5.2.1. Single flow scenario

In this scenario, a single flow from the source (node A) to the sink (node E) having an inelastic average

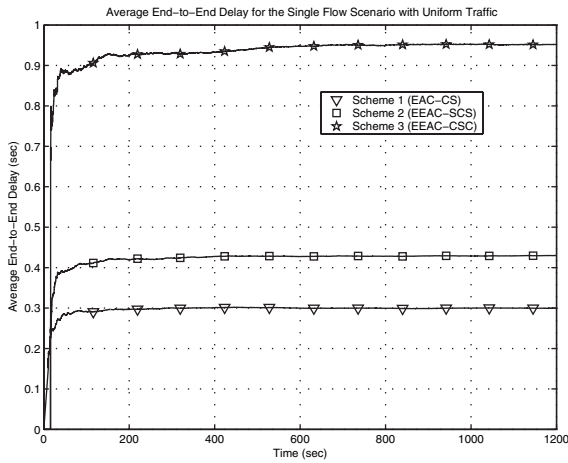


Fig. 4. Average end-to-end delay of the three schemes for the single flow scenario with uniform traffic.

end-to-end delay requirement of 950 ms is considered. The performance of the three different types of QoS provisioning schemes; that is, Scheme 1 (EEAC-CS), Scheme 2 (EEAC-SCS), and Scheme 3 (EEAC-CSC) are evaluated and compared. Since real-time data flows are assumed for the flows from the source to the sink, they always use service class 0 under the EEAC-CS scheme.

Figures 4 and 5 demonstrate the performance of the three schemes, in terms of the average end-to-end delay and average end-to-end power consumption, when the total number of packets that can be generated by the source in a time slot is uniformly distributed with a maximum of four packets/time slot.

Specifically, Figure 4 demonstrates that all the three schemes can satisfy the inelastic end-to-end average

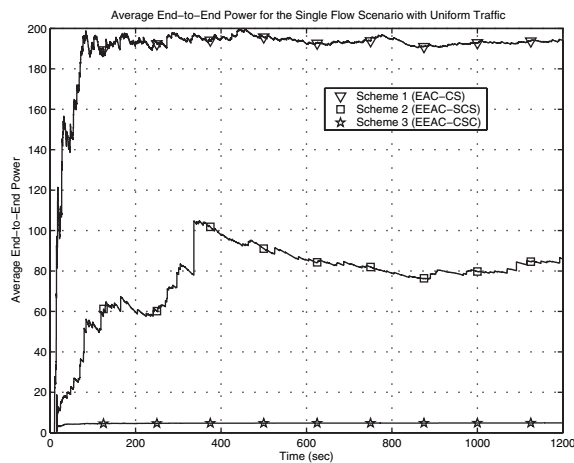


Fig. 5. Average end-to-end power consumption of the three schemes for the single flow scenario with uniform traffic.

delay requirement. EEAC-CSC (Scheme 3) results in longer end-to-end delay than EEAC-SCS (Scheme 2) (respecting however the delay requirement) because it tries to make use of all possible combinations of service classes, therefore, loading all the service classes. As a result, a packet may experience longer delay in EEAC-CSC scheme. For the same reason, EEAC-CS (Scheme 1) results in the smallest average end-to-end delay.

There is, however, a trade-off between the achievable end-to-end delay and the corresponding power consumption, as demonstrated by Figure 5 that compares the average end-to-end power consumed by the flow under consideration for the three types of QoS provisioning schemes. Specifically, Scheme 3 (EEAC-CSC) results in the lowest power consumption, whereas Scheme 1 (EEAC-CS) leads to the highest power consumption. This clearly demonstrates that the proposed approach of implementing the *service vector* concept in wireless ad-hoc networks in combination with the described delay bounded formulation of the multi-flow wireless scheduling discipline, results in significant power savings over the conventional static service mapping (EEAC-CS) scheme, as well as over the single class of service (EEAC-SCS) scheme. In other words, the method proposed in this paper enables the *service vector* concept, which was originally developed for wire-line networks, to enhance the end-to-end QoS in wireless ad hoc networks.

The impact of the *On-Off* traffic on the performance of the corresponding QoS provisioning schemes is demonstrated in the following Figures 6 and 7. In this scenario, we assume that the *On* state and the *Off* state

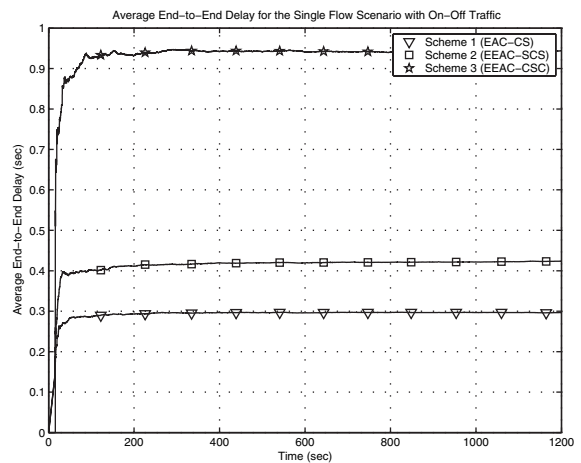


Fig. 6. Average end-to-end delay of the three schemes for the single flow scenario with *On-Off* traffic.

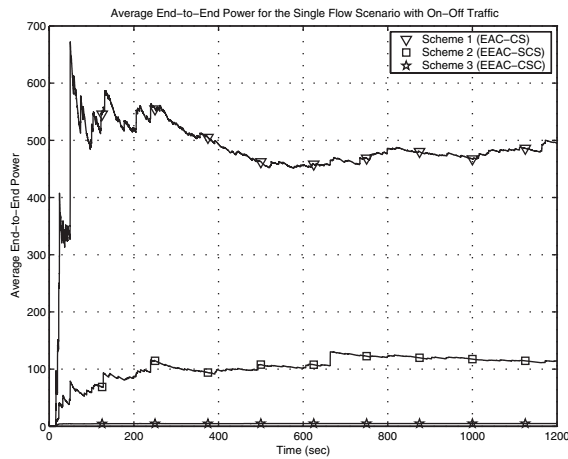


Fig. 7. Average end-to-end power consumption of the three schemes for the single flow scenario with *On-Off* traffic.

are equally likely, and four packets are generated while in the *On* state. From these figures, we also confirm that with respect to the power consumption the EEAC-CSC scheme clearly outperforms the other two schemes for this traffic arrival pattern (*On-Off* arrivals), while still satisfies the average end-to-end delay requirements. For all the three schemes, *On-Off* arrivals result in higher power consumption than their uniform arrival distribution counterparts. The reason for this is attributed to the fact that the *On-Off* arrival process requires the highest transmit power at any delay in an AWGN channel among all arrival processes with the same average and finite maximum arrival rate [13].

5.2.2. Two flow scenario

In this scenario, two different types of flows generated by the source are considered; that is, *Type 1* which has an end-to-end average delay bound of 950 ms, and *Type 2* with an average end-to-end delay bound of 750 ms. Fifty per cent of the traffic generated by the source is *Type 1* and the remaining is *Type 2*. The cross traffic remains the same as in the single flow case. As mentioned earlier, the objective of this scenario—where two flows with different QoS requirements are considered—is to better demonstrate the service differentiation capabilities of our proposed mechanism.

More specifically, Figures 8 and 9 demonstrate the performance of the three QoS provisioning schemes, in terms of the average end-to-end delay and average end-to-end power consumption, where the total number of packets that can be generated by the source

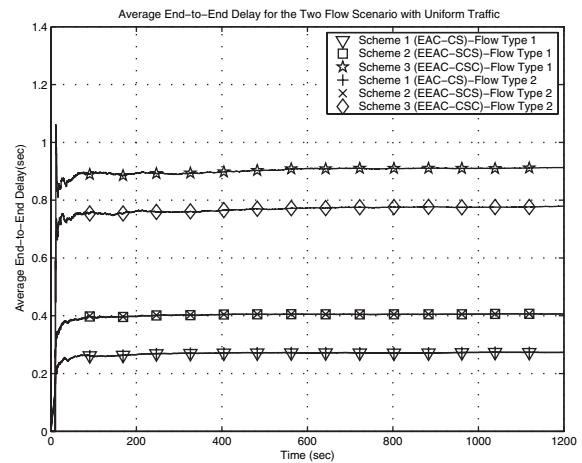


Fig. 8. Average end-to-end delay of the three schemes for the two flow scenario with uniform traffic.

in a time slot is uniformly distributed with a maximum of four packets/time slot. As illustrated in these figures, EEAC-CSC scheme (Scheme 3) is the only scheme that can provide service differentiation, by providing different quality to each one of the two different flows according to their requirements. On the other hand, EEAC-SCS and EEAC-CS schemes are unable to provide this differentiation since they map these two flows to the same service vector. In other words, the integrated method proposed in this paper enables finer QoS granularity both in terms of the average end-to-end delay and average end-to-end power consumption.

Similarly, Figure 10 presents the end-to-end average delay for the two flows when *On-Off* traffic is

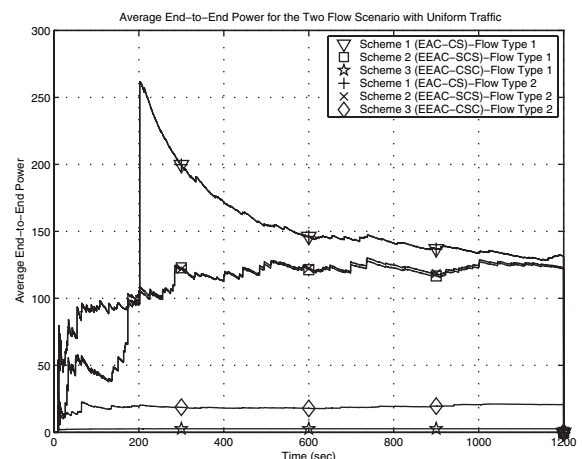


Fig. 9. Average end-to-end power consumption of the three schemes for the two flow scenario with uniform traffic.

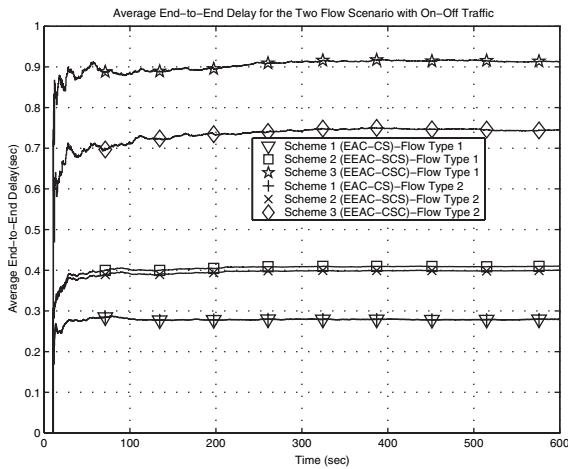


Fig. 10. Average end-to-end delay of the three schemes for the two flow scenario with *On-Off* traffic.

generated by the source with the same characteristics as the *On-Off* traffic in the single-flow case. Furthermore, Figures 11a–c illustrate the average power consumption for the two flows, under each one of the three QoS provisioning schemes when *On-Off* traffic is considered. While Scheme 3 leads to smaller power consumption for flow *Type 1*, which has a less strict delay bound than flow *Type 2*, and hence is able to differentiate between these two flows, Scheme 1 and 2 result in the same average end-to-end power consumption for these two flows having different QoS requirements and therefore fail to provide this differentiation. From these results, the capability of our proposed approach in providing finer QoS granularity both in terms of average end-to-end delay and power consumption becomes evident. For the same reasoning as before, the power consumption for each scheme

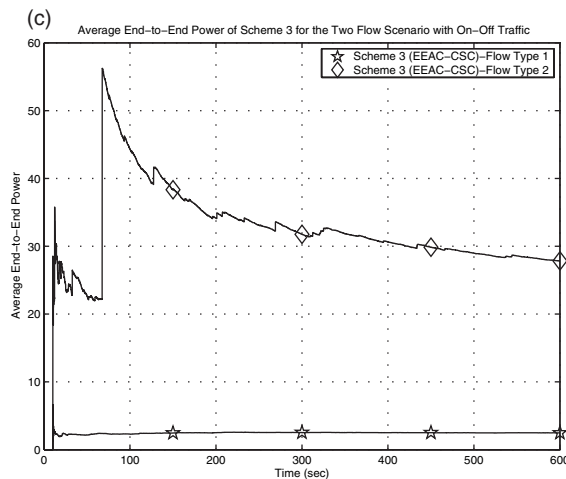
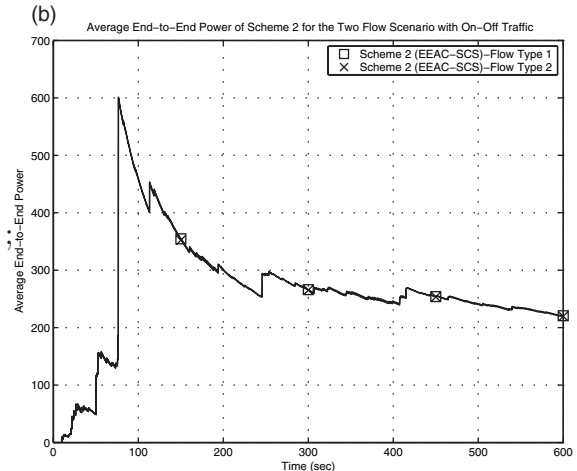
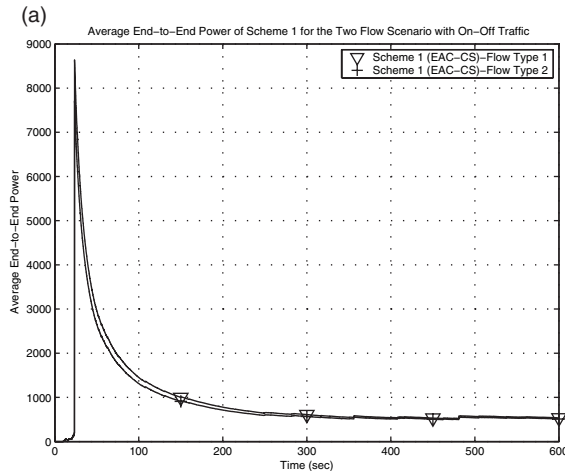


Fig. 11. (a) Average end-to-end power consumption under Scheme 1 for the two flow scenario with *On-Off* traffic. (b) Average end-to-end power consumption under Scheme 2 for the two flow scenario with *On-Off* traffic. (c) Average end-to-end power consumption under Scheme 3 for the two flow scenario with *On-Off* traffic.

under the *On-Off* traffic arrival pattern is higher than the corresponding ones under the uniform arrival distribution counterparts.

On the other hand, it can also be observed from Figures 11a–c that the average end-to-end power consumption for times below 100 ms present high variations when the source traffic is *On-Off*, especially when compared with the corresponding ones under the uniform arrival distribution counterparts. This is due to the high variance and bursty nature of *On-Off* traffic. The variance of the uniform and *On-Off* traffic source under consideration here is $\text{var}(\text{uniform source}) = M_{\text{src}}^2/12 + M_{\text{src}}/6$ and $\text{var}(\text{On-Off source}) = M_{\text{src}}^2/4$, where M_{src} denotes the maximum number of packets that can be generated by the source in a time slot. Consequently, $\text{var}(\text{On-Off source}) \geq \text{var}(\text{uniform source}) \forall M_{\text{src}} \geq 1$. When the traffic source is *On-Off*, initially the number of packets in the buffers increases more rapidly than the uniform traffic source case. Therefore, when the traffic source is *On-Off*, the power consumption at the initial phases of the data flow is much more than the uniform traffic source case. Furthermore, the variation of average end-to-end power in Scheme 1 (Figure 11a) is larger than the variation of Scheme 2 (Figure 11b), which is in turn larger than the variation of average end-to-end power in Scheme 3. The reason for this is that since Scheme 1 chooses class 0 at all the nodes along the path and class 0 is the service class with the most stringent delay requirement, at the initial phases of the data flow the scheduler's flow choice is most of the time in favor of the class 0 buffer in order to meet the average delay requirement of this service class. This situation leads to an initial peak in the average end-to-end power consumption of this service class. Due to the same reasoning, the average power consumption of class 1 buffer also has an initial peak, which is smaller than the peak of class 0 buffer, since service class 1 has a less stringent delay requirement than class 0. Since Scheme 2 uses class 1 buffers along the entire path, this peak is directly reflected in its average end-to-end power consumption. On the other hand, since class 2 which has the least stringent delay bound, is permitted to be utilized under Scheme 3 the initial peak of average end-to-end power consumption of Scheme 3 for both flows is less than the peaks of Schemes 1 and 2. These results also demonstrate that our proposed methodology (Scheme 3, EEAC-CSC) results in less variance in average end-to-end power consumption, and hence more stable QoS performance at the initial phases of the data flow, compared to the performance of Scheme 1 and 2.

6. Conclusions and Future Work

A cross-layer architecture that achieves significant power savings, while enhancing the end-to-end QoS provisioning and granularity in wireless ad hoc networks, is considered in this paper. An integrated scheme, which utilizes both link layer delay-bounded power efficient multi-user wireless scheduling and the network layer concept of *service vector*, is introduced and evaluated. It has been demonstrated, through modeling and simulation, that in wireless networks significant power savings, as well as enhanced QoS granularity and service differentiation can be achieved, based on the proposed approach. Furthermore, the distributed nature of our proposed scheme makes it especially suitable for wireless ad hoc networking environments. The impact of various traffic arrival distributions as well as flows with different QoS requirements, on the performance of the proposed strategy has also been investigated.

Due to the inefficiencies and implementation complexity of the optimal multi-user wireless scheduler, suboptimum scheduler which can operate only in AWGN channels has been utilized in this study. Extending this suboptimum scheduler to take fading into account would be of high practical and research importance in order to investigate the implications of fading on the performance of the *service vector* scheme. Furthermore, the probing process can be utilized to gather additional information regarding the channel performance such as fading coefficients, which are usually unknown to the end user device.

Acknowledgements

This work has been supported in part by the National Science Foundation under Grant 0435250.

References

1. Ramanathan R, Redi J. A brief overview of ad hoc networks: challenges and directions. *IEEE Communications Magazine* 2002; **40**(5): 20–22.
2. Chakrabarti S, Mishra A. QoS issues in ad hoc wireless networks. *IEEE Communications Magazine* 2001; **39**(2): 142–148.
3. Braden R, Clark D, Shenker S. Integrated services in the Internet architecture: an overview. *RFC1633*, 1994.
4. Blake S, Black D, Calson M, Davies E, Wang Z, Weiss W. An architecture for differentiated services. *RFC2475*, 1998.
5. Yang J, Ye J, Papavassiliou S, Ansari N. A flexible and distributed architecture for adaptive end-to-end QoS provisioning in next generation networks. *IEEE Journal on Selected Areas in Communications* 2005; **23**(2): 321–333.

6. Yang J, Ye J, Papavassiliou S. Enhancing end-to-end QoS granularity in Diffserv networks via service vector and explicit endpoint admission control. *IEEE Proceedings on Communications* 2004; **151**(1): 77–81.
7. Fang JC, Rao RR. Flow control for end-to-end delay and power constrained wireless multihop networks. *Proceedings of IEEE Military Communications Conference* 2004; pp. 487–492.
8. Nichols K, Carpenter B. Definition of differentiated services per domain behaviors and rules for their specification. *RFC3086*, 2001.
9. Wang KC, Ramanathan P. End-to-end delay assurances in multihop wireless local area networks. *Proceedings of IEEE Global Telecommunications Conference* 2003; 2962–2966.
10. Dovrolis C, Stiliadis D, Ramanathan P. Proportional differentiated services: delay differentiation and packet scheduling. *IEEE/ACM Transactions on Networking* 2002; **10**(1): 12–26.
11. Dovrolis C, Ramanathan P. Dynamic class selection and class provisioning in proportional differentiated services. *Computer Communications Journal* 2003; **26**(3): 204–221.
12. Cisco, *Networking Technology Handbook* 2002; ch. 49, Cisco Press.
13. Berry RA, Gallager RG. Communication over fading channels with delay constraints. *IEEE Transactions on Information Theory* 2002; **48**(5): 1135–1149.
14. Collins BE, Cruz RL. Transmission policies for time varying channels with average delay constraints. *Proceedings of Allerton International Conference on Communication, Control and Computing* 1999; pp. 709–717.
15. Rajan D, Sabharwal A, Aazhang B. Delay-bounded packet scheduling of bursty traffic over wireless channels. *IEEE Transactions on Information Theory* 2004; **50**(1): 125–144.
16. Rajan D. Power efficient transmission policies for multimedia traffic over wireless channels. *Ph.D. thesis* 2002; Rice University.
17. Floyd S, Jacobson V. Random early detection for congestion avoidance. *IEEE/ACM Transactions on Networking* 1993; **1**(4): 397–413.

Authors' Biographies



Didem Gozupak received the B.S. degree (high honors) in Telecommunications Engineering from Sabanci University, Istanbul, Turkey, in 2004 and the MS degree in electrical engineering from New Jersey Institute of Technology, NJ, USA, in 2005. During her graduate studies, she had been a research assistant in the Broadband, Mobile, and Wireless Networking

Laboratory, NJIT. Currently she is working as an R&D engineer for Argela Technologies, Istanbul, Turkey. Her main research interests are in the fields of computer and communication networks, in particular, QoS provisioning mechanisms, cross layer design, and wireless ad hoc network applications.



Symeon Papavassiliou (S'92-M'96) received the Diploma in Electrical Engineering from the National Technical University of Athens, Greece, in 1990 and the MSc and PhD degrees in

electrical engineering from Polytechnic University, Brooklyn, New York, in 1992 and 1995, respectively. Currently he is with the Faculty of Electrical and Computer Engineering Department, National Technical University of Athens. From 1995 to 1999, Dr Papavassiliou was a senior technical staff member at AT&T Laboratories in Middletown, New Jersey, and in August 1999 he joined the Electrical and Computer Engineering Department at the New Jersey Institute of Technology (NJIT), where he was an Associate Professor. Dr Papavassiliou was awarded the Best Paper Award in INFOCOM'94, the AT&T Division Recognition and Achievement Award in 1997, and the National Science Foundation (NSF) Career Award in 2003. Dr Papavassiliou has an established record of publications in his field of expertise, with more than one hundred technical journal and conference published papers. His main research interests lie in the areas of computer and communication networks with emphasis on wireless communications and high-speed networks.



Nirwan Ansari received the B.S.E.E. degree (summa cum laude) from the New Jersey Institute of Technology (NJIT), Newark, NJ, in 1982, the M.S.E.E. degree from the University of Michigan, Ann Arbor, MI, in 1983, and the Ph.D. degree from Purdue University, West Lafayette, IN, in 1988. He joined the Department of Electrical and Computer Engineering,

NJIT, as an Assistant Professor in 1988, and has been a Full Professor since 1997. He authored *Computational Intelligence for Optimization* (Kluwer, 1997) with E.S.H. Hou and translated into Chinese in 2000, and co-edited *Neural Networks in Telecommunications* (Kluwer, 1994) with B. Yuhua. He is a Senior Technical Editor of the *IEEE Communications Magazine*, and also serves on the editorial board of *Computer Communications*, the *ETRI Journal*, and the *Journal of Computing and Information Technology*. His current research focuses on various aspects of broadband networks and multimedia communications. He has also contributed over 90 refereed journal articles, plus numerous conference papers and book chapters. He initiated (as the General Chair) the First IEEE International Conference on Information Technology: Research and Education (ITRE2003), was instrumental, while serving as its Chapter Chair, in rejuvenating the North Jersey Chapter of the IEEE Communications Society which received the 1996 Chapter of the Year Award and a 2003 Chapter Achievement Award, served as Chair of the IEEE North Jersey Section and in the IEEE Region 1 Board of Governors during 2001–2002, and has been serving in various IEEE committees including as TPC Chair/Vice-chair of several conferences. He was the 1998 recipient of the NJIT Excellence Teaching Award in Graduate Instruction, and a 1999 IEEE Region 1 Award. He is frequently invited to deliver keynote addresses, tutorials, and talks. He has been selected as an IEEE Communications Society Distinguished Lecturer (2006–2007).