A Proportional Jitter Scheduling and Buffer Management Algorithm for Differentiated Services Networks

Mohammad Hossein Yaghmaee
Computer Department, Faculty of Engineering, Ferdowsi University of Mashad (FUM), Iran
hyaghmae@ferdowsi.um.ac.ir

Gh .Khojasteh Tousi
Azad University of Mashad, Iran
ghazale_khojaste@yahoo.com

Abstract- **This paper proposes a method for scheduling and buffer management in differentiated services networks which is based on JoBS algorithm. JoBS differentiates traffic classes base on delay and loss rate. Furthermore, it supports both relative and absolute constraints. Nowadays with the expansion of Internet traffic and its diversified service requests, the focus is on requirement of new traffic used on the Internet. The proposed method focuses on packet scheduling based on the jitter (variance of delay), since new traffics that will be mostly used in the next generation Internet are more susceptible to variance of delay. Drawing on concepts of jitter and the difference between scheduling base on jitter and delay, this paper analyzes the effectiveness of the proposed method by simulation examples that are presented here. Results show that proposed method can proportionally differentiated classes based on delay, loss rate and jitter. Also, it fulfills the absolute conditions which are defined by absolute parameters. This paper concludes with the suggestion that this method comparing to the JoBS can proportionally differentiate classes based on more parameters and its result of differentiation is better.**

Index Term- **Buffer Management, Scheduling, Quality-of-Service, Differentiated Services, JoBS, Proportional Differentiated Services**

I. INTRODUCTION

Quality of Service (QoS) refers to a set of rules or techniques that help the network administrators use the available network resources optimally to manage the effects of congestion and to treat the applications according to their needs. The primary goal of QoS is to provide priority including dedicated bandwidth, controlled jitter and latency (required by some real-time and interactive traffic), and improved loss characteristics. Also important is making sure that providing priority for one or more flows does not make other flows fail [1].

In the last few years, the growth of the Internet and the use of new services such as e-business, voice over IP (VoIP)[2] and multimedia applications has risen the need to support QoS requirements and to accommodate different service levels. The differentiated services architecture (DiffServ) [3] allows providing quality of service to users. The major DiffServ premise is that individual flows with similar QoS requirements can be aggregated in larger traffic sets and identified as classes. All packets in each traffic class, receive the same 'forwarding behavior' in routers [4]. Two directions exist in the DiffServ architecture: the absolute and the relative.

In absolute DiffServ [5], an admission control scheme is used [6] to provide QoS guarantees as absolute bounds of specific QoS parameters such as bandwidth, packet transfer delay, packet loss rate, or packet delay variation (jitter). A connection request is rejected if sufficient resources are not available in the network so as to provide the desirable assurances. End to end performance requires passive or active monitoring procedures along a specific connection before its establishment and throughout its lifetime. Thus, for any admitted user the appropriate resources are reserved and the performance level of the connection is assured [7].

The relative DiffServ model [8] provides QoS guarantees per class in reference to guarantees given to other classes. The only assurance from the network is that higher classes receive better service treatment than lower classes. QoS parameter values for a connection depend on the current network load since there is no admission control and resource reservation mechanism. Relative service differentiation is a simple and easily deployed approach compared to the absolute differentiation service [7].

This paper is divided into six sections including the introduction. Second section gives an overview proportional Differentiated services. The JoBS algorithm is described in section 3. The forth section specifies the proposed algorithm. Section 5 presents simulation scenarios to evaluate the effectiveness of the proposed algorithm and compares it with JoBS. The final section concludes by discussing that proposed algorithm can proportionally differentiate traffic classes based on jitter, loss rate and even delay.

II. PROPTIONAL DIFFERENTIATED SERVICE

Proposals for relative per class DiffServ QoS define service differentiation qualitatively [9-10], in terms that higher classes receive lower delays and losses from lower classes. Specifically research effort has focused on a qualitative relative differentiation scheme named proportional DiffServ [11-12], which controls the ratios of delays or loss rates of successive priority classes in order to be constant.

In the following paragraph, a generic description of the proportional differentiation model as described in [11] is given. Suppose that $\bar{q}_i(t,t+\tau)$ is a performance measure for class i in the time interval $(t,t+\tau)$, where $\tau > 0$ is the monitoring timescale. The proportional differentiation model imposes constraints of the following form for all pairs of classes and for all time intervals $(t,t+\tau)$ in which both $\bar{q}_i(t,t+\tau)$ and $\bar{q}_j(t,t+\tau)$ are defined, the following equation is satisfied:

$$\frac{\bar{q}_i(t,t+\tau)}{\bar{q}_j(t,t+\tau)} = \frac{c_i}{c_j} \tag{1}$$

where $c_1 < c_2 < \ldots < c_N$ are the generic Quality Differentiation Parameters (QDPs). The basic idea is that, even though the actual quality level of each class will vary with the class loads, the quality ratio between classes will remain fixed and controllable by the network operator, independent of the class loads [8].

The proportional differentiation model can be applied in three contexts, proportional delay differentiation [11], proportional loss rate differentiation [12] and proportional jitter differentiation model [13-15].

In the case of proportional delay differentiation [11], defined $\bar{q}_i(t,t+\tau) = 1/\bar{d}_i(t,t+\tau)$ where $\bar{d}_i(t,t+\tau)$ is the average queuing delay of the class i packets that departed in the time interval $(t,t+\tau)$. If there are no such packets, $\bar{d}_i(t,t+\tau)$ is not defined. The proportional delay differentiation model states that for all pairs of classes and for all time intervals (t, t + τ) in which both $\bar{d}_i(t,t+\tau)$ and $\bar{d}_j(t,t+\tau)$ are defined, the following equation is satisfied:

$$\frac{\bar{d}_i(t,t+\tau)}{\bar{d}_j(t,t+\tau)} = \frac{\delta_i}{\delta_j} \tag{2}$$

where the parameters $\{\delta_i\}$ are the Delay Differentiation Parameters (DDPs), being ordered as $\delta_1 > \delta_2 > \ldots > \delta_N$.

In the case of proportional loss rate differentiation [12], defined $\bar{q}_i(t,t+\tau) = 1/\bar{l}_i(t,t+\tau)$ where $\bar{l}_i(t,t+\tau)$ is the fraction of class i packets that were backlogged at time t or arrived during the interval (t, t + τ), and were dropped in this time interval. In this case, the proportional loss rate differentiation takes the form:

$$\frac{\bar{l}_i(t,t+\tau)}{\bar{l}_j(t,t+\tau)} = \frac{\sigma_i}{\sigma_j} \tag{3}$$

where the parameters $\{\sigma_i\}$ are the Loss rate Differentiation Parameters (LDPs), being ordered as $\sigma_1 > \sigma_2 > \ldots > \sigma_N$. In the case of proportional jitter differentiation [14], defined $\bar{q}_i(t,t+\tau) = 1/\bar{j}_i(t,t+\tau)$ where $\bar{j}_i(t,t+\tau)$ is the average jitter of the class i packets that departed in the time interval $(t,t+\tau)$. In this case, the proportional jitter differentiation takes the form:

$$\frac{\bar{j}_i(t,t+\tau)}{\bar{j}_j(t,t+\tau)} = \frac{\Delta_i}{\Delta_j} \tag{4}$$

where the parameters $\{\Delta_i\}$ are the jitter differentiation parameters (JDPs), being ordered as $\Delta_1 > \Delta_2 > \ldots > \Delta_N$.

Several scheduler and dropper are presented about proportional differentiated services [11-21].

This paper, proposes a new algorithm for scheduling and buffer management in IP Diffserv network. With the expansion of Internet traffic and its diversified service requests, the focus is on requirement of new traffic used on the Internet. In the proposed algorithm scheduling packets are based on jitter. The proposed method that based on JoBS, proportionally differentiates classes based on jitter and loss rate. Because of jitter is very close to delay, result in this method shows that besides differentiate traffic classes that are proportionate to JDP and LDP , these classes can be separated based on delay as well.

III. JOINT SCHEDULING AND BUFFER MANAGEMENT ALGORITHMS

Although joint scheduling and buffer management can improve the existing algorithms, but this subject is rarely being given attention. The number of these algorithms in contrast to other algorithms that can check only delay or only loss rate is very limited.

Equivalent Differentiated Service (EDS) [22] can be deployed incrementally and provides different but equal services. This scheme differentiates a number of N traffic classes on loss and delay without giving on absolute better service to anyone. EDS uses WTP [11] as scheduler and PLR [12] as dropper. In [23], three schedulers based on the relative differentiated service model have been proposed. For packet scheduling with only delay, only loss and both delay and loss differentiation. As described in [23], they adopt the (m, k) model and Distance Based Priority (DBP) scheduling algorithm [24] for scheduling real-time streams. Their method designed for multicasting. JoBS [25] is the powerful algorithm that manages dropping and scheduling in one step. Since proposed algorithm in this paper is based on JoBS, the following paragraph will discuss about the JoBS algorithm. As described in [25], JoBS assumes per-class buffering of arriving traffic and serves traffic from the same class in a First-Come-First-Served order. JoBS allocates to each traffic class a guaranteed service rate. The service rate guarantees are adjusted over time and may be changed as often as after each traffic arrival. Within the context of JoBS, there is no admission control and no policing of traffic. The set of relative or absolute performance requirements are given to the JoBS algorithms as a set of per-class QoS constraints. The set of constraints given to JoBS can be any mix of relative and absolute constraints. The JoBS algorithm operates as follows.

For every arrival, JoBS makes a prediction on the delays of the backlogged traffic, and modifies the service rates so that all QoS and system constraints will be met. If changing the service rates is not sufficient for meeting all constraints, JoBS will drop either the arrival or it will drop queued traffic.

IV. The Proposed Algorithm

In the proposed algorithm which is based on the well known JoBS algorithm, two types of QoS constraints (absolute and relative constraints) are defined. The QoS constraints specify the variance of delay and loss rate. So, four different categories are defined as follows:

- Absolute Jitter Constraints (AJC): This kind of constraint enforces the algorithm to schedule the packets in a way that jitter of a class that AJC is defined for it, in all situations must be smaller or equal to the defined value.
- Relative Jitter Constraints (RJC): This constraint is a definer of proportional jitter differentiation between the classes. Algorithm has to schedule the packets in a way that the ratio of jitter between different classes becomes equal with the RJC value.
- Absolute Loss Constraints (ALC): This kind of constraint is so effective in the buffer management. The loss rate of packets in a class which the ALC constraint is defined for it, must be always equal or less than the defined value.
- Relative Loss Constraints (RLC): This constraint specifies the proportional loss differentiation between the classes.

The difference between the proposed algorithm and JoBS algorithm is in definition of AJC and RJC. The proposed algorithm does the scheduling of packets according to their jitters but in the buffer management acts as JoBS. Because of the traffics (voice, video) that nowadays are so common in the Internet are susceptible to the variance of delay (jitter), the proposed algorithm is focused on the jitter problem. But, as jitter and delay are so close together, focus on the jitter problem will automatically overshadow the delay problem.

In the following sub-section, at first jitter, loss rate and the way of calculating them will be defined. Then the algorithm's operation will be evaluated in two states: packets entrance and packets exit.

A. jitter

Jitter of one packet in a queue is the difference of queuing delay of this packet and the preceding packet in this class [13]. Suppose d_i^k is the delay of k'th packet in class i and j_i^k is the jitter of this packet. The j_i^k is calculated by the flowing formula:

$$j_i^k = \left| d_i^k - d_i^{k-1} \right| \tag{7}$$

Entrance time and exit time of some packets and calculation of jitter are illustrated in Fig. 1.As the JDPs announce the jitter ratio of two classes, therefore the jitter of a class will be defined. Usually in a class, instead of calculating the immediate jitter, the average jitter is calculated. In this way, in short periods of time such as $[t-\tau, t]$ the average jitter of all packets (j_i^{avg}) that got serviced in this time period, is calculated as below:

$$j_i^{avg} = \frac{\sum_{k=1}^{s_i} j_i^k}{s_i} \tag{8}$$

where s_i is the number of serviced packets in time interval $[t-\tau, t]$ and j_i^k is the jitter of k'th packet in this time interval of class i.

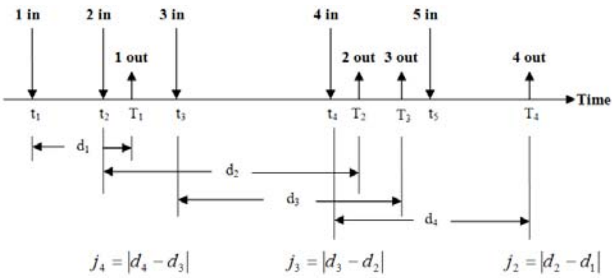


Figure1. Ingress time and egress time of some packet of one class and jitter calculating

B. Loss Rate

Loss rate of a traffic class is the ratio of total dropped packets to total arrived packets. If $a_i(t)$ and $l_i(t)$ denote the traffic arrivals and amount of dropped traffic from class i that measured in time interval $[t-\tau, t]$, loss rate of class i in this time interval can be calculated as below:

$$lossrate_i = \frac{\int_{t-\tau}^{t} l_i(x)dx}{\int_{t-\tau}^{t} a_i(x)dx} \tag{9}$$

C. The Algorithm's Operation at the Time of Traffic Entrance

With the entrance of each packet to the router, at first, the amount of the existent free space in buffer is evaluated and in the lack of free space drop traffic must be acted. Therefore, this algorithm enters the decision making steps about the drop. All the performed levels of this section are as like as JoBS [25] algorithm that will be explained more summarily.

According to the average of loss rate in each class from the beginning till now and also with the consideration of ALC and RLC which are defined for each class, the suitable class will be recognized by algorithm. Accordance to the operator's opinion deletion operation can de done with one of the two methods: droptail or dropfront.

After performing the above actions, the algorithm for all the classes must adjust the rate of service. The difference between the proposed algorithm with JoBS algorithm is in this level. Adjusting the rate of service in all the classes is performed according to the defined RJC and the current situation of each class.

For all the classes which have the backlogged packets, assume that the packet in front of the queue at this moment is planning to exit the queue. So, by this assumption the average of the predicted jitter for each of the classes is obtained by the following equation:

$$j_i^p = \frac{j_i^{avg} \times s_i(t) + j_i^x}{s_i(t) + 1} \quad (10)$$

where j_i^{avg} is the average jitter of class i that was calculated before without considering the new packet. $s_i(t)$ is the number of packets that participate in calculating average jitter. j_i^x is jitter of head-of-line packets of class i.

After calculating predicted jitter of each class (j_i^p), the average of multiplying j_i^p and RJC of each class is obtained by the following equation:

$$Mean = \frac{\sum_{i=1}^{numberofclass} j_i^p \times \Delta_i}{numberofclass} \quad (11)$$

where as mentioned before, Δ_i is the jitter differentiation parameter. As described in [25], since the link bandwidth is fixed, so the summation of service rate of all classes is also fixed. Hence, in this stage to reach the desired quality of service, the service rate of all classes is adjusted by decreasing the service rate of the classes that require less and increase the service rate of classes that require more. To perform this action the distance of average (mean variable) with the current service rate in each class is obtained as below:

$$Diff_i = Mean - (j_i^p \times \Delta_i) \quad (12)$$

Two situations will be occurred; if the obtained amount of variable Diff for a class is negative, this class requires more service rate. Else this class can meet the requirement of QoS with less service rate.

Now by decreasing the service rate of classes in second situation and increasing the service rate of classes in first situation, this shortage will be compensated. The amount of decrement or increment's rate of service in each class is calculated by a function that exists in the context of the JoBS's source code and are called *DecreaseServiceRate()* and *IncreaseServiceRate(),* respectively. The pseudo code of adjusting the service rate is as follows:

Receive (packet)
Queue in corresponding class;
For each class i;

$$j_i^p = \frac{j_i^{avg} \times s_i(t) + j_i^x}{s_i(t) + 1}$$

$$Mean = \frac{\sum_{i=1}^{numberofclass} j_i^p \times \Delta_i}{numberofclass}$$

For each class i;
$Diff_i = Mean - (j_i^p \times \Delta_i)$
If $Diff_i < 0$
Do IncreaseServiceRate ( );
Else
Do DecreaseServiceRate ( );

D. The Algorithm's Operation at the Time of Traffic Exit

When the traffic exits from router, two operations will be performed: selecting a class for servicing its head-of-line packet and updating the average jitter of desired class. In the first operation, earliest the classes that AJC is defined for them will be considered. For these classes, difference between jitter of head-of-line packet and defined AJC will be calculated and maximum of this value will determined as below:

$$j = \max(J_i^{head-of-line} - AJC_i) \quad (13)$$

Selected-class=i

If value of j be positive, the head-of-line packet of selected class will be serviced, otherwise the classes that RJC is defined for them will be considered. Then, according to the defined service rate for each class, the number of serviced packet of each class can be predicted. By comparing the amount of packets that each class is serviced till now and the amount of predicted packets, a class with the most distance between these two amounts will be selected. After selecting the class and also exiting the packet from the class, the amount of average jitter in this class will be updated by considering the jitter of exited packet. The average amount of jitter of each class is calculated in short periods of time and these amounts will be referred to as the short term jitter of each class. The pseudo code of selecting a class and updating jitter will be as follows:

For each class i that AJC is defined:
$j = \max(J_i^{head-of-line} - AJC_i)$
Selected-class=i
If $j > 0$ *then*
Head-of-line packet serviced from class j at time t
Else
{
For each class i that RJC is defined:
Packet drop from class j at time t:
$j = \arg\max_i \left(Rout_i^p - Rout_i\right)$
}
$pkt_j{+}{+}$

$$j_j^{avg} = \frac{j_j^{avg} \times (pkt_j - 1) + j_j^{head-of-line}}{pkt_j}$$

V. Simulation Results

To evaluate the performance of the proposed algorithm, we consider three different experiments. At the first and the second experiment only the relative parameters are considered while at the third experiment both the relative and absolute parameters are considered simultaneously. At the first experiment, the type of traffic is common for all the traffic classes. But in the second experiment for each traffic class one kind of traffic is considered. In all three experiments, the load distribution is asymmetric. We used NS2 network simulator developed by National Berkeley Laboratory [26] as the simulation package.

A. Experiment 1

In the first experiment, we simulate a network which its topology is shown in Fig. 2. This network topology was used in [25] to evaluate the performance of JoBS algorithm. In this test, just the relative differentiated between the traffic classes is being observed. Network topology consist of four routers that connected by three 45 Mbps links. Sources and sinks are connected to the routers by independent 100 Mbps links. Each 45 Mbps link has a propagation delay of 3 ms, and each 100 Mbps link has a propagation delay of 1 ms. There exist four different traffic classes.

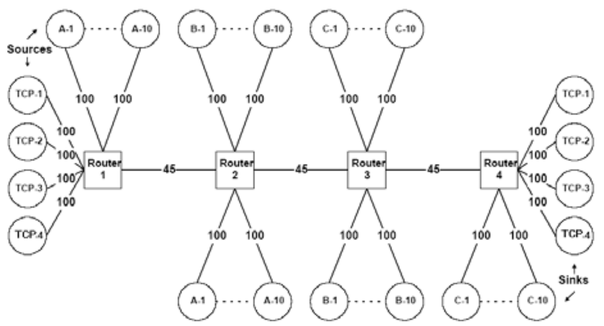


Figure 2. Network topology used in the simulation

The composition of the traffic mix is given in table 1. The service guarantees of these classes are as below:

Class-4 Jitter ≈ 2 × Class-3 Jitter
Class-3 Jitter ≈ 2 × Class-2 Jitter
Class-2 Jitter ≈ 2 × Class-1 Jitter
Class-4 Loss rate ≈ 2 × Class-3 Loss rate
Class-3 Loss rate ≈ 2 × Class-2 Loss rate
Class-2 Loss rate ≈ 2 × Class-1 Loss rate

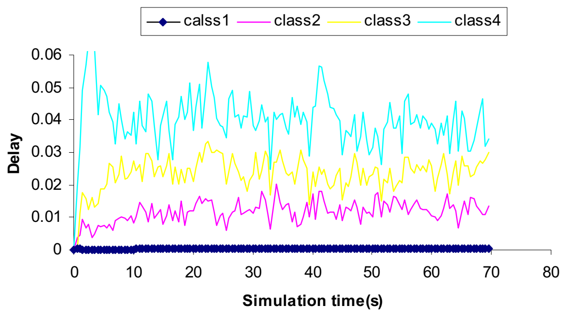
Cross-traffic flows (denoted by A-1,…,C-10) start transmitting at time t = 0 s. The flows TCP-1,TCP-2, TCP-3 and TCP-4 start transmitting at time t = 10 s. All flows consist of packets with a fixed size of 500 bytes, and the simulation time is set to 70s. The offered load is asymmetric. Classes1,2,3 and 4 contribute 10%,20%,30% and 40% of the aggregate cross-traffic, respectively.

Per-Hop Per-class QoS

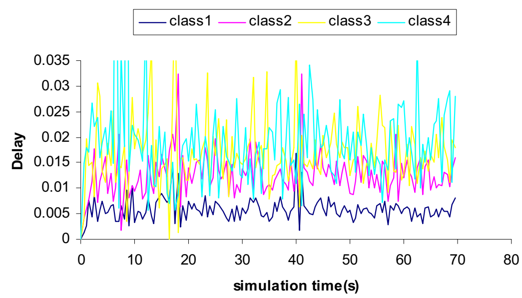
The per-class queuing delay and per-class loss rate at each of the routers that are generated by the proposed algorithm and JoBS algorithms, are shown in Fig. 3 and 4. In Fig. 3, for both algorithms and for Router1 in the network topology, the queuing delay of different classes is plotted versus simulation time. Based on results shown in this figure, it is clear that in all network's routers the proposed algorithm in comparison with JoBS algorithm has more authority to differentiate the delay of traffic classes. Also the oscillations in the related diagrams of the proposed algorithm seem to be less.

TABLE I. Traffic mix of experiment 1. Traffic mix for flows B-1,..., B-10 and C-1,...C-10 is identical to the traffic mix described here for flows A-1,...,A-10.

Flow	Class	Protocol	Traffic	On	Off
TCP-1	1	TCP	Greedy	N/A	N/A
TCP-2	2	TCP	Greedy	N/A	N/A
TCP-3	3	TCP	Greedy	N/A	N/A
TCP-4	4	TCP	Greedy	N/A	N/A
A-1	1	TCP	On-Off	1000pkts	200ms
A-2,A-3	2	TCP	On-Off	1000pkts	200ms
A-4,A-5,A-6	3	TCP	On-Off	1000pkts	200ms
A-7,A-8,A-9,A-10	4	TCP	On-Off	1000pkts	200ms



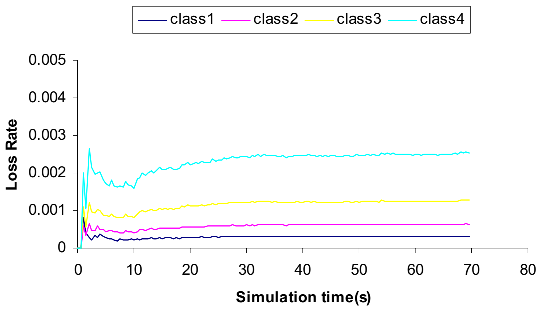
(a) Router1- Delay of Classes (Proposed algorithm)



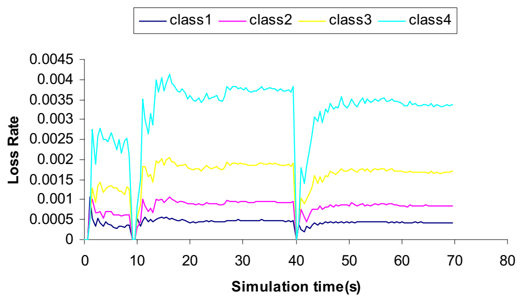
(b) Router1- Delay of Classes (JoBS algorithm)

Figure 3. Experiment 1: delay of classes for proposed algorithm and JoBS

In Fig. 4, for both proposed algorithm and JoBS algorithm and for Router2 in the network topology, the loss rate of different classes is plotted versus simulation time. As the buffer management of our proposed algorithm is similar to JoBS algorithm, the diagrams are nearly congruent to each other.



(a) Router 2- Loss Rate (Proposed algorithm)



(b) Router 2- Loss Rate (JoBS algorithm)

Figure 4.Experiment 1: loss rate of classes for proposed algorithm and JoBS algorithm

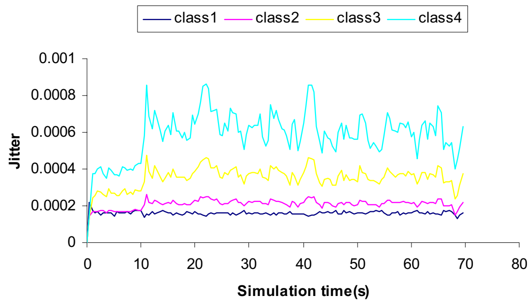
In Fig. 5, for both the proposed algorithm and JoBS algorithm and for Router3 in the network topology, the jitter of different classes is plotted versus simulation time. The results shown in these diagrams prove that JoBS algorithm doesn't have any capability to differentiate the traffic classes according to the jitter. But the proposed algorithm does the packets scheduling in a way that the jitter ratio of different traffic classes be compatible with the defined RJC.

End-to-end Per-flow QoS

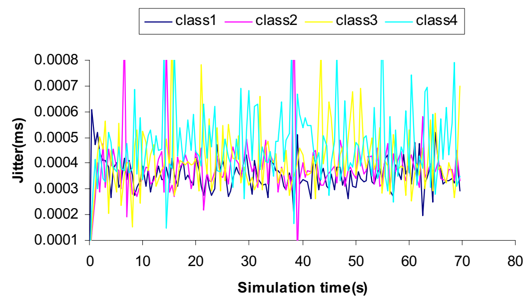
In this section we present the end-to-end delay measurements for the flows TCP-1, TCP-2, TCP-3 and TCP-4. Fig. 6, for both algorithms, shows the end-to-end delay of flows. By looking at Fig. 6(b), it can be recognized that the end to end delay of traffic flows in JoBS algorithm is not completely differentiated. But, in the proposed algorithm this differentiation is perfectly specified and obvious. Furthermore, in the proposed algorithm the end to end delay has less oscillation and also the average delay in each class is less than JoBS algorithm.

B. Experiment 2

In the second experiment, we simulate a network with a topology like Fig. 2 but the types of traffic and the number of sources and sinks have been changed. In the new topology there are 40 UDP sources (v-1,…,v-40) that are generated VoIP traffic, 40 UDP sources (R-1,…,R-40) that are generated RealAudio traffic, one UDP source that is generate pareto on/off traffic and one TCP source that is generated FTP traffic. The number of routers and link's characteristics are similar to experiment1.

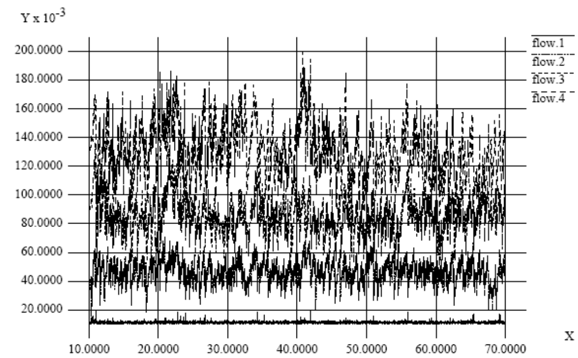


(a) Router 3- Jitter of Classes (Proposed algorithm)

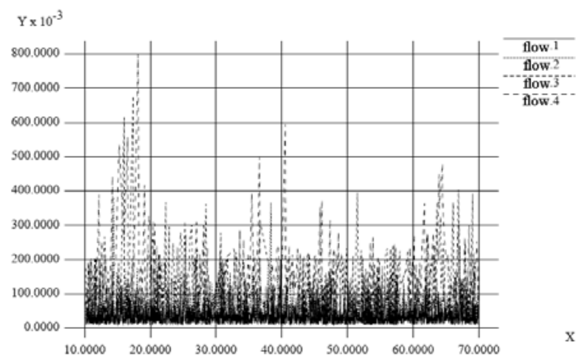


(b) Router 3- Jitter of Classes (JoBS algorithm)

Figure 5.Experiment 1:jitter of classes for proposed algorithm and JoBS algorithm



(a) Proposed algorithm



(b) JoBS

Figure 6. End-to-end delay of flows

Like expriment1, there are four traffic classes. The composition of the traffic mix is given in table 2 and the service guarantees of these classes are as below:

Class-4 Jitter ≈ 2 × Class-3 Jitter
Class-3 Jitter ≈ 2 × Class-2 Jitter
Class-2 Jitter ≈ 2 × Class-1 Jitter
Class-4 Loss rate ≈ 2 × Class-3 Loss rate
Class-3 Loss rate ≈ 2 × Class-2 Loss rate
Class-2 Loss rate ≈ 2 × Class-1 Loss rate

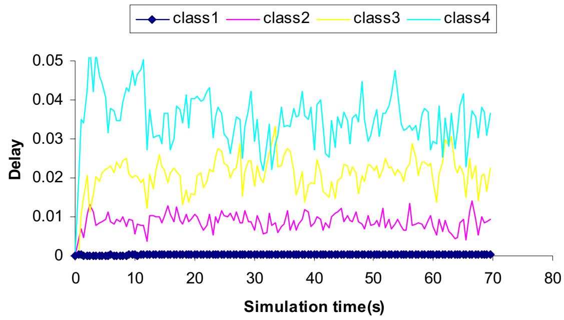
Cross-traffic flows (denoted by A-1,…, C-10) start transmitting at time t = 0 s. The flows VoIP, RealAudio, UDP and TCP start transmitting at time t = 10 s. All flows consist of packets with a fixed size of 500 bytes, and the simulated time is set to 70s. Like expriment1, in this case the offered load is also asymmetric. Class1 contributes 10% of the aggregate cross-traffic, class2 contributes 20%, class3 contributes 30% and class4 contributes 40%.

The per-class queuing delay and per-class loss rate at each of the routers that are generated by proposed algorithm and JoBS algorithm, were evaluated. Due to lake of space, only the results of Rouer1 will be given. In Fig. 7, for both proposed algorithm and JoBS algorithm, the queuing delay, loss rate and jitter of different classes are plotted versus simulation time. Based on results shown in this figure, it is clear that the proposed algorithm in comparison with JoBS algorithm has more authority to differentiate the delay, loss rate and jitter of the traffic classes.

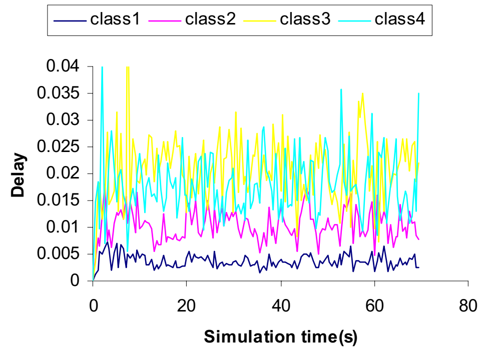
TABLE II. TRAFFIC MIX OF EXPERIMENT 2. TRAFFIC MIX FOR FLOWS B-1,...,B-10 AND C-1,...C-10 IS IDENTICAL TO THE TRAFFIC MIX DESCRIBED HERE FOR FLOWS A-1,...,A-10.

Flow	Class	Protocol	Traffic	On	Off	Shape
V-1,…, V-40	1	UDP	Exponential	1587 ms	1004 ms	N/A
R-1,…, R-40	2	UDP	Real Audio	50ms	1800000 ms	N/A
UDP	3	UDP	Pareto	10ms	10ms	1.5
TCP	4	TCP	Greedy	N/A	N/A	N/A
A-1	1	TCP	Exponential on-of	1000 pkts	200 ms	N/A
A-2, A-3	2	TCP	Exponential on-of	1000 pkts	200 ms	N/A
A-4, A-5, A-6	3	TCP	Exponential on-of	1000 pkts	200 ms	N/A
A-7, A-8, A-9, A-10	4	TCP	Exponential on-of	1000 pkts	200 ms	N/A

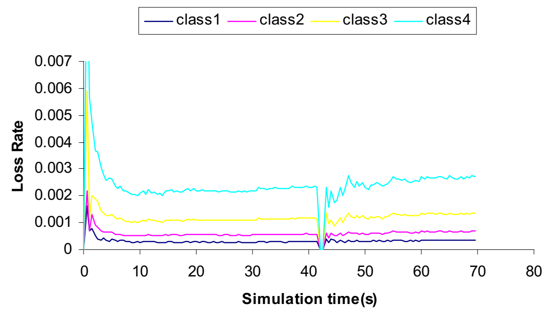
To evaluate the end-to-end delay of each flow, we consider that all of VoIP traffic streams belonging to class1 and also all of RealAudio traffic streams belonging to class2, each one produce one flow. Fig. 8 shows end-to-end delay of four flows by this assumption. It can be easily seen that the proposed algorithm has better performance than traditional JoBS algorithm.



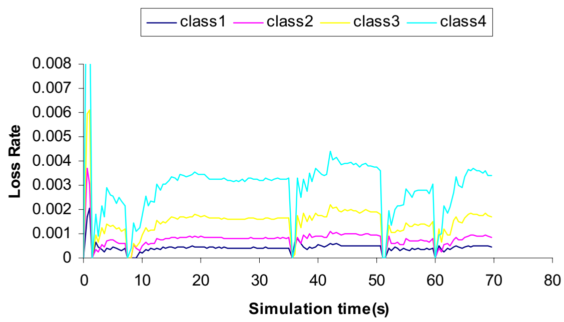
(a) Router 1- Delay of Classes (Proposed algorithm)



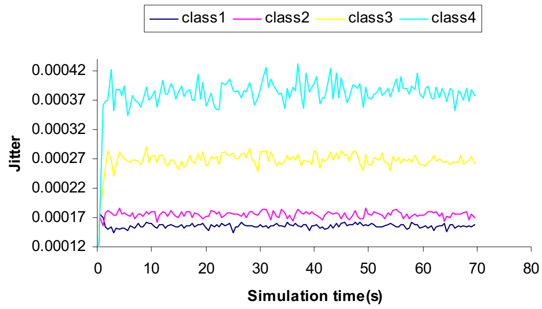
(b) Router 1- Delay of Classes (JoBS algorithm)



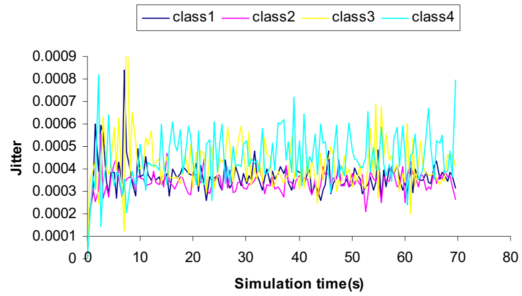
(c) Router 1- Loss Rate (Proposed algorithm)



(d) Router 1- Loss Rate (JoBS algorithm)

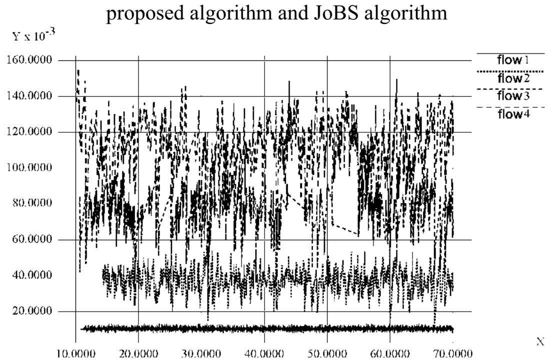


(e) Router1- Jitter of Classes (Proposed algorithm)

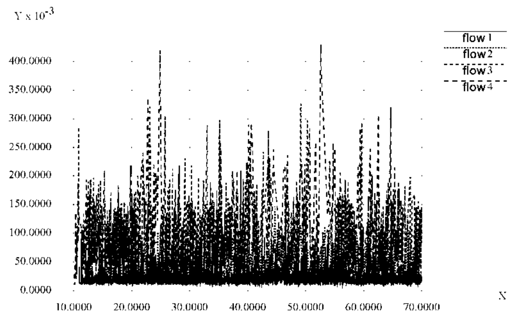


(f) Router 1- Jitter of Classes (JoBS algorithm)

Figure 7. Experiment 2: delay, loss rate and jitter of classes for both proposed algorithm and JoBS algorithm



(a) Proposed algorithm



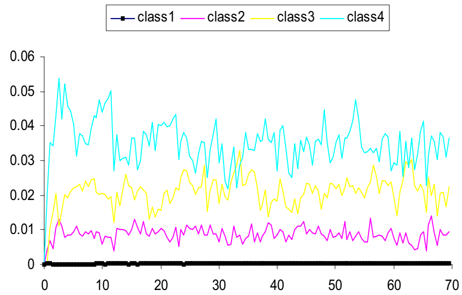
(b) JoBS

Figure 8. End-to-end delay of flows

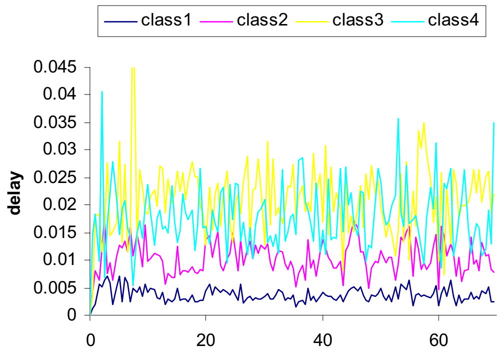
C. Experiment 3

In this experiment, both relative and absolute parameters will be considered simultaneously. Network topology and traffic classes are similar to experiment 2.

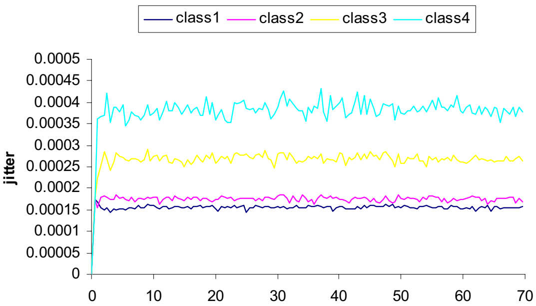
The diagrams of Fig. 9, for both proposed algorithm and JoBS algorithm, show the delay and jitter of traffic classes. The service guarantees of these classes are like to experiment 2. In this case no absolute constraints are defined.



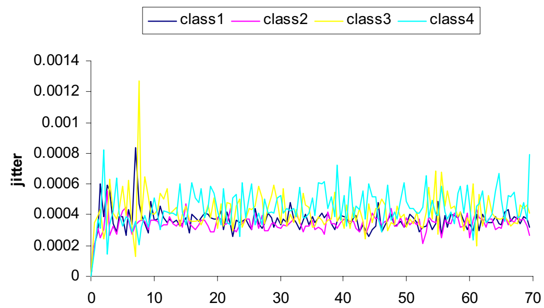
(a) Delay of Classes (Proposed algorithm)



(b) Delay of Classes (JoBS algorithm)



(c) Jitter of Classes (Proposed algorithm)



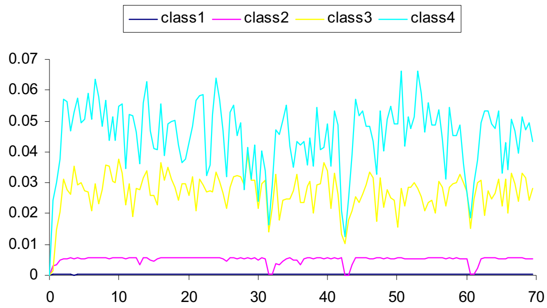
(d) Jitter of Classes (JoBS algorithm)

Figure 9. Experiment 3: Delay and jitter of classes without absolute constraints

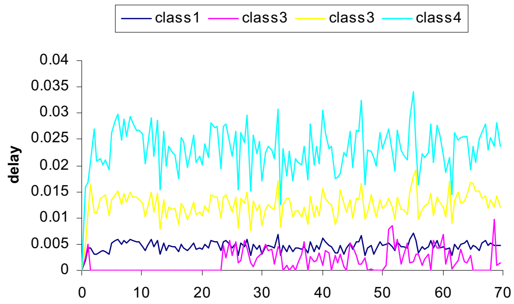
In the next experiment, we define an ADC such as below:

Class-2 delay ≈ 12 ms

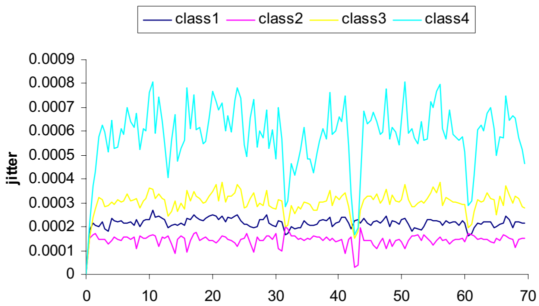
In Fig. 10, for both proposed algorithm and JoBS algorithm, the delay and jitter of traffic classes are plotted versus simulation time.



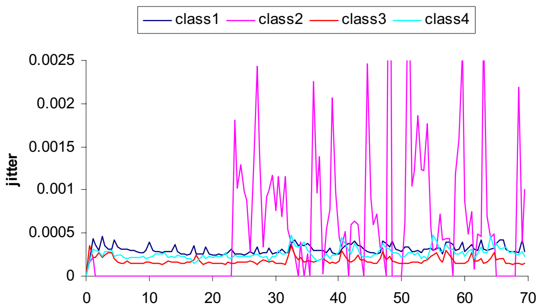
(a) Delay of Classes (Proposed algorithm)



(b) Delay of Classes(JoBS algorithm)



(c) Jitter of Classes (Proposed algorithm)



(d) Jitter of Classes (JoBS algorithm)

Figure10: Experiment 3: delay and jitter of classes with ADC for class 2, delay of class2 ≈ 12 ms

Based on the results shown in Fig. 9-10, it can be seen that both algorithms can guarantee the delay constraints, but the proposed algorithm can schedule the packets in a way that both absolute and relative delay and jitter constraints can be satisfied.

VI. CONCLUSION

This paper proposed an algorithm for proportional differentiated services which can satisfy three different constraints including delay, loss rate and jitter. The proposed algorithm uses the relative and absolute constraints which are defined by operator. Scheduling and buffer management are in a way to supply the desired conditions. The proposed algorithm in addition of the used constraints in JoBS, uses two other constraints namely AJC and RJC. Furthermore, it can differentiate the traffic classes according to the delay, jitter and loss rate proportionally. To show the effectiveness of the proposed algorithm and comparing it with JoBS, different experiments were performed. At the first and the second experiments only the relative parameters were considered while at the third experiment the relative and absolute parameters were considered simultaneously. In all three experiments load distribution was asymmetric. The simulation results showed that the proposed algorithm is able to differentiate the traffic classes according to the jitter, loss rate and even delay. Also, the proposed algorithm fulfills the absolute conditions which are defined by absolute parameters. Comparing the results that are collected from different experiments showed that the proposed algorithm can act better than traditional JoBS algorithm.

REFERENCES

[1] Y. Bernet, The Complementary Roles of RSVP and Differentiated Services in the Full-Service QoS Network, *IEEE Communications Magazine*, vol. 38, no. 2, February 2000, pp. 154–162.

[2] C. Chuah, L. Subramarian, and R. Katz, Furies: A Scalable Framework for Traffic Policing and Admission Control, May 2001, U.C Berkeley Technical Report No. UCB/CSD-01-1144.

[3] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W.Weiss, An Architecture for Differentiated Services, IETF RFC 2475, December 1998.

[4] M.K. Leung, J.C. Lui, and D.K. Yau, Characterization and Performance Evaluation for Proportional Delay Differentiated Services Proc. Int'l Conf. Network Protocols, pp. 295-304, Nov. 2000.

[5] B. Teitelbaum, QBone Architecture (v1.0). Internet2 QoS Working Group Draft, http://www.internet2.edu/qos/wg/papers/qbArch/1.0/draft i2-qbone-arch-1.0.html, Aug. 1999

[6] I. Stoika and H. Zhang, LIRA: An approach for Service Differentiation in the Internet, In Proceedings NOSS-DAV, 1998

[7] Angelos Michalas, Paraskevi Fafali, Malamati Louta, Vassilios Loumos, Proportional Delay Differentiation Employing the CBQ Service Discipline, Telecommunications, 2003. ConTEL 2003.

[8] C. Dovrolis and Parmesh Ramanathan, A Case for Relative Differentiated Services and the Proportional.

Differentiation Model. In IEEE Network, 13(5):26-34, September 1999 (special issue on Integrated and Differentiated Services in the Internet).
[9] A. Odlyzko, Paris Metro Pricing: The Minimalist Differentiated Services Solution, In Proceedings IEEE/IFIP International Workshop on Quality of Service, June 1999.
[10] D. Clark and W. Fang, Explicit allocation of best-effort packet delivery service, In IEEE/ACM Transactions on Networking, Vol. 6 (4), Aug.1998.
[11] C. Dovrolis, D. Stiliadis, and P. Ramanathan, Proportional differentiated services: delay differentiation and packet scheduling, in Proc. ACM SIGCOMM 1999, Cambridge MA, September 1999, pp. 109–120.
[12] C. Dovrolis and P. Ramanathan, Proportional differentiated services, part II: Loss rate differentiation and packet dropping, in *Proc. International Workshop on Quality of Service (IWQoS 2000)*, Pittsburgh, PA, June 2000, pp. 52–61.
[13] T. Ngo-Quynh, H. Karl, A. Wolisz and K. Rebensburg, Relative jitter packet scheduling for Differentiated Services, Proc. Of 9th IFIP working conference on performance modeling and evaluation of ATM&IP networks IFIP ATM&IP, pp. 139-151, 2001.
[14] T. Ngo-Quynh, H. Karl, A. Wolisz and K. Rebensburg, New scheduling algorithm for providing proportional jitter in differentiated services network, Proceedings of IST mobile communication and wireless telecommunications summit, Thessaloniki, Greece, June, 2002.
[15] T. Ngo-Quynh, H. Karl, A. Wolisz and K. Rebensburg, Using only Proportional Jitter Scheduling at the boundary of a Differentiated Service Network: simple and efficient, In Proc. of 2nd European Conf. on Universal Multi service Networks (ECUMN), pp.116-123, Colmar, France, April 2002.
[16] Y. Moret and S. Fdida, A proportional queue control mechanism to provide differentiated services, In *Proceedings of the International Symposium on Computer and Information Systems (ISCIS)*, pages 17–24, Belek, Turkey, October 1998.
[17] T. Nandagopal, N. Venkitaraman, R. Sivakumar, and V. Barg havan, Delay differentiation and adaptation in core stateless networks, In Proceedings of IEEE INFOCOM 2000, pp. 421–430, Tel-Aviv, Israel, April 2000.
[18] H. T. Ngin and C. K. Tham, Achieving proportional delay differentiation efficiency, Proceedings of the 10th IEEE International Conference on Networks (Icon 2002), pp. 169-174, Singapore, August, 2002.
[19] C. Dovrolis, Proportional differentiated services for the Internet, PhD thesis, University of Wisconsin-Madison, December 2000.
[20] U. Bodin, A. Jonsson, and O. Schelen, On creating proportional loss differentiation: predictability and performance, In Proceedings of IWQoS 2001, pp. 372–386, Karlsruhe, Germany, June 2001.
[21] A. Kumar, J. Kaur and H. Vin, End-to-end Proportional Loss Differentiation, Technical Report. TR-01-33, University of Texas, February 2001.
[22] B. Gaidioz, P. Primet, EDS: A new scalable Service Differentiation Architecture for Internet, Proceedings of International Symposium on Computer Communication (ISCC) 2002, pp. 777-782, Taormina, Italy, July 2002, IEEE.
[23] A. D. Striegel and G. Manimaran, Packet Scheduling with Delay and Loss differentiation, Computer Communications, vol.25, no.1, pp.21-31, Jan. 2002.
[24] M. Hamdaoui and P. Ramanathan, A dynamic priority assignment technique for streams with (m,k)-Firm Deadlines, IEEE Trans. Computers, vol.44, no.12, pp.1443-1451, Dec. 1995.
[25] J. Liebeherr and N. Christin, "JoBS: Joint buffer management and scheduling for differentiated services", In Proceedings of IWQoS 2001, pp. 404–418, Karlsruhe, Germany, June 2001.
[26] NS-2 Network simulator http://www.isi.edu/nsnam/ns/

Mohammad Hossein Yaghmaee. Mohammad Hossein Yaghmaee was born on July 1971 in Mashad, Iran. He received his B.S. degree in Communication Engineering from Sharif University of Technology, Tehran, Iran in 1993, and M.S. degree in communication engineering from Tehran Polytechnic (Amirkabir) University of Technology in 1995. He received his Ph.D degree in communication engineering from Tehran Polytechnic (Amirkabir) University of Technology in 2000. He has been a computer network engineer with several networking projects in Iran Telecommunication Research Center (ITRC) since 1992.Currently he is associated professor in comuter department of Ferdowsi University of Mashad. He is author of 3 books and more than 60 international papers. His research interests are in traffic and congestion control, high speed networks including ATM and MPLS, Quality of Services (QoS) , wireless sensor networks and fuzzy logic control.

Ghazale Khojasteh Tousi. Ghazale Khojasteh Tousi was born on January 1977 in Mashad, Iran. She received her B.S. degree in Computer Engineering from Azad University of Mashad, Iran in 1999 and her M.S. degree from Azad University of Mashad, Iran in 2007. Her research interests are in computer networking and Internet.