Transmitting Packets Using Hybrid Scheduling

Roxana Stănică*, Emil Petre**

**University of Craiova, Romania (e-mail: roxana_batm@yahoo.com)*
***University of Craiova, Romania (e-mail: epetre@automation.ucv.ro)*

Abstract: Quality of Service (QoS) controls the traffic congestion by assigning priorities to the transmitted packets. QoS queuing policies can protect bandwidth for important categories of applications, or specifically limit the bandwidth associated with less critical traffic. In this paper, four new Hybrid scheduling commands are proposed. The new commands combine Strict Priority (SP) and Weighted Round Robin (WRR) scheduling. The transmission of packets is done using eight queues, and each queue can be controlled by one of the Hybrid commands.

Keywords: priority, packet, hybrid scheduling, command

1. INTRODUCTION

Quality of Service (QoS) refers to resource reservation control mechanisms rather than the achieved service quality. QoS can provide different priority [7] to different applications, users, or data flows, or can guarantee a certain level of performance to a data flow. For example, a required bit rate, delay, jitter, packet dropping probability and/or bit error rate may be guaranteed [1]. High QoS is often confused with a high level of performance or achieved service quality, for example high bit rate, low latency, and low bit error probability.

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 [10]. Also important is making sure that providing priority for one or more flows does not make other flows fail. QoS Monitoring involves tracing levels (parameters) for an application and compares them with those required.

1.1 QoS Architecture

Fig. 1 [9] shows the traffic flow during the QoS process.

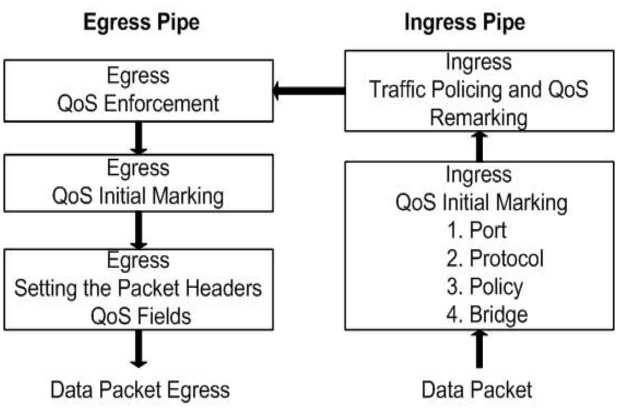

Fig. 1. Standard QoS architecture

On ingress (the right or permission to enter) pipe, the traffic is remarked according to the rate limit (detailed in Rate Limit section).

On egress (the right or permission to leave) pipe, traffic is distributed into eight priority queues according to internal priority and drop precedence (color). The traffic can be transmitted with one of the queuing algorithm: Strict Priority (SP), Weighted Round Robin (WRR), or the Hybrid scheduling algorithm [8] (detailed in Traffic Scheduling section).

The QoS architecture (see Fig. 1) is based on the following:

- At the network edge (ingress), the packet is assigned to a QoS service. The service is assigned based on the packet header information (if the packet is trusted) or on the ingress port configuration (if the packet is untrusted).
- The QoS service defines the packet internal QoS handling (Class of Service - CoS and drop precedence - Color) and optionally the packet external QoS marking, through either the 802.1p User Priority and/or the IP header DSCP field (detailed in Packet’s Attributes section).
- A switch may modify the assigned CoS if a packet stream exceeds the configured profile. In this case, the packet may be dropped or reassigned to a lower CoS.

1.2 QoS Process

QoS processing [9] is divided (see Fig. 1) in ingress and egress pipes:

- (Egress) QoS Enforcement: uses eight egress queue-priorities per port. Congestion avoidance and congestion resolution techniques are used to provide the required service.
- (Egress) QoS Initial Marking: associates every packet with a set of QoS attributes. All types of

packets - data, control, and mirrored to analyzer port - are subject to egress QoS initial marking.

- (Egress) Setting the Packet Header QoS Fields: the packet header 802.1p User Priority and IP-DSCP (Differentiated Services Code Point) are defined or modified.
    - IEEE 802.1p standard [10, 8]: specifies a priority value between 0 and 7 that can be used by Quality of Service (QoS) protocol to differentiate traffic.
    - Differentiated Services (DiffServ) [11, 8]: the network tries to deliver a particular service based on QoS specified by each packet.
- (Ingress) Traffic Policing and QoS Remarking: if it is enabled on a policy-based traffic flow, and if the packet is classified as data, the given flow is measured according to a configurable rate limit that classifies packets as either in-profile or out-of-profile.
- (Ingress) QoS Initial Marking: associates every packet classified as data with a set of QoS attributes that determine the QoS processing by subsequent stages (shown in Fig. 1).

1.3 Traffic Analysis

To configure QoS, the types of traffic [5] have to be analyzed and the ports relative bandwidth demands have to be established. Also, the supported applications' sensitivity must be evaluated to:

- Delay: the transit time an application needs from the ingress point to the egress point of the network.
- Jitter: the measure of delay variation between consecutive packets for a given traffic flow. Jitter has a pronunced effect on real-tine, delay-sensitive applications such as voice and video.
- Packet loss: the routers may fail to deliver some packets if they arrive when their buffers are already full. Some, none, or all of the packets may be dropped, depending on the state of the network.

The traffic types are:

- Voice: demands small amounts of bandwidth. However, the bandwidth must be constant and predictable because voice applications are sensitive to latency (inter-packet delay) and jitter.
- Video: similar to voice application but requires larger bandwidth, depending on the encoding. Some applications can transmit large amounts of data for multiple streams in one spike or burst, causing the switch to buffer significant amounts of sent video-stream data.
- Database (DB): does not demand significant bandwidth and is tolerant to delay. Therefore, it requires minimum bandwidth and can be set to use lower priority.
- Web-browsing: cannot be generalized into a single category. Most browser-based applications have an asymmetric dataflow (small dataflow from the client's browser and large dataflow from the server to the client). An exception to this pattern might be created by some Java-based applications. Web-based applications are generally tolerant of latency, jitter, and some packet loss.
- File server: has the greatest demand on bandwidth, although it is tolerant to latency, jitter, and some packet loss, depending on the network operating system and the use of TCP (Transmission Control Protocol) or UDP (User Datagram Protocol).

1.4 Tail-Drop Configuration

Congestion avoidance techniques ensure the monitoring of traffic tasks in a network. One of these techniques is the *tail-drop* mechanism. When the queues reach their maximum length, this mechanism drops the recently received packets until the congestion is eliminated and the queue is no longer full. Tail-drop treats all traffic flows equally and does not differentiate between classes of service. Up to eight tail-drop profiles can be configured.

1.5 Packet's Attributes

Every packet has assigned an initial set of QoS attributes. This assignment is done using several QoS markers contained in the ingress pipe. This ingress pipe also contains a QoS remarker that can modify the initial QoS attributes. The packet's QoS attributes are:

- *QoS Precedence*: a switch can incorporate multiple QoS markers operating in a sequence. As a result, a later marker overrides an earlier QoS attribute assignment.
- *QoS Profile Index*: is used as a direct index, from 0 to 127, into the global QoS Profile table. Each entry in this table contains the following set of attributes:
    - TC/FC (Traffic Class/Forwarding Class): one of the following traffic classes can be used: be (Best-Effort), 12 (Low-2), af (Assured), 11 (Low-1), h2 (High-2), ef (Expedited), h1 (High-1) or nc (Network Control).
    - DP (Drop Precedence):within each class, packets have a drop precedence (high, medium or low). If congestion occurs within a class, the packets with the higher drop precedence are discarded first.
    - UP: if the packet's QoS attribute is set to <Modify UP> and the packet is received untagged, this field is the value used in the packet 802.1p User Priority field. If the switch receives a tagged packet, the existing User Priority is modified with this value.

      An Ethernet packet that contains a VLAN ID (Virtual Local Area Network Identifier) is a tagged packet. Conversely, an Ethernet packet with no VLAN ID is an untagged packet.
    - DSCP: if the packet's QoS attribute is set to <Modify DSCP> and the packet is IPv4 or IPv6,

this field is the value used to modify the packet IP-DSCP field.
    - QoS profiles are used for all types of services. In this paper, the valid range is <1-8>.

- *Modify DSCP*: enables packet IP-DSCP field when the packet egresses the switch:
    - 0 = Packet IP-DSCP field is preserved;
    - 1 = Packet IP-DSCP field is modified to the <DSCP> value of the QoS profile entry for the packet QoS Profile Index.
- *Modify User Priority*: enables packet 802.1p User Priority field when the packet egresses the switch:
    - 0 = Packet User Priority field is preserved;
    - 1 = Packet User Priority field is modified to the <UP> value of the QoS Profile entry for the packet QoS Profile Index.
- *Default User Priority*: is assigned by the ingress port configuration, only when the <Modify UP> is cleared and the packets are received untagged.

2. TRAFFIC SCHEDULING

Traffic Scheduling allows the control of packets transmission, based on priorities assigned to those packets. Congestion management determines the creation of queues, the assignment of packets to the queues (based on the packet's classification), and scheduling of the packets in a queue (for transmission).

The packets are scheduled for transmission according to their assigned priority and their queuing algorithm. The switch determines the order of packets transmission by controlling which packets are placed in which queue and the order in which the queues are serviced.

The QoS traffic behaviour can be controlled by selecting the queuing algorithm to be applied to the outbound queues (eight queues are used).

Three queuing algorithms can be used:

- Strict Priority (SP)
- Weighted Round-Robin (WRR)
- Hybrid scheduling that combines SP and WRR. New Hybrid queuing commands are proposed.

2.1 Strict Priority (SP)

SP [9] provides preferential treatment for high-priority traffic, making sure that mission-critical traffic gets priority treatment.

SP Algorithm:

- The high-ranking queue q8 is serviced first until it is empty,

  then
- The low-ranking queue q7 is serviced and so on, down to q1.

SP provides a faster response time for high-priority traffic than other methods of queuing. The SP algorithm guarantees a fixed portion of available bandwidth to an application (e.g., interactive multimedia applications), possibly at the expense of less critical traffic.

When selecting SP, the lower priority traffic is often denied in favor of high-priority traffic. In the worst case, lower priority traffic is never transmitted. However, these scenarios can be avoided by using the rate limit (detailed in Rate Limit section) to control high-priority traffic rate. Rate limit controls congestion on service provider networks, and ensures proper use of bandwidth resources.

The disadvantage of using SP relates to the way the strict priority treats queues. High-priority packets are always processed before those of less priority. Medium-priority packets are always processed before low-priority packets. If the amount of high-priority traffic is great, other queues might never empty, leading to worse performance for the low-priority and medium-priority traffic.

Fig. 2 illustrates the SP process in four-queue architecture. The incoming packets are classified with a high-priority transmission (e.g., 1), and are transmitted through a queue with high-priority (e.g., q8).

2.2 Weighted Round Robin (WRR)

WRR [9] is a scheduling algorithm that cycles through the queues. A weighting factor determines how many bytes of data the system delivers from each queue before moving to the next queue. WRR scheduling prevents the low-priority queues from being completely neglected during periods of high-priority traffic.

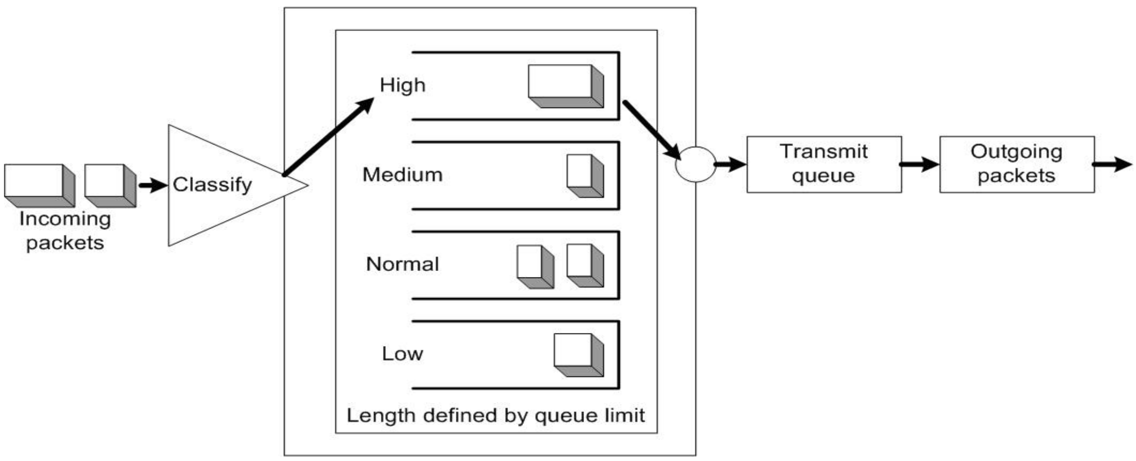

Fig. 2. Strict Priority queuing

By using this scheduling, low-priority queues have the opportunity to transmit packets even though the high-priority queues are not empty.

WRR Algorithm:

- The packets in the queue are sent until the number of bytes to be transmitted exceeds the bandwidth or until the queue is empty (only then WRR moves to the next queue).
- If a queue is empty, the switch sends packets from the next queue that has packets to send.
- If a packet's length exceeds the queue's allowed bandwidth, the packet is still transmitted during its time slot, but its quota is overdrawn so next time it receives a smaller allocation.

This algorithm guarantees a minimum bandwidth for each queue, but allows the minimum to be exceeded if one or more of the port's other queues are idle.

Fig. 3 illustrates the WRR queuing in four-queue architecture. The incoming packets are classified with an average transmission priority (e.g., 3), and are transmitted through a queue with medium priority (e.g., q4).

2.3 Hybrid Algorithm

The Hybrid scheduling algorithm [8] combines the SP and WRR algorithms. This algorithm ensures delivery of critical packets while there is no priority assigned to the packets. (For detailed information on the new Hybrid commands, refer to the New Scheduling Commands section.)

Hybrid Algorithm:

- The queues with high-priority are serviced with SP;
- The remaining queues are serviced in accordance with WRR, after the high-priority queues are empty.

Hybrid queuing guarantees immediate delivery of packets from high-ranking queues while avoiding lowest-ranking queues.

3. RATE LIMIT

Rate limiting is performed by policing (discarding excess packets), queuing (delaying packets in transit) or congestion control (manipulating protocol's congestion mechanism). Traffic congestion, caused by heavy network traffic, can cause incoming packets to be dropped. The rate-limit command, which control the traffic behaviour, can be found in [8]. This command applies and configures a rate limit for all ports of the switch.

A traffic rate limiter monitors the incoming traffic by:

- forwarding conforming traffic (within the predefined rate);
- dropping non-conforming traffic or marking traffic.

3.1 Single Rate Three Color Marker (RFC 2697)

The Single Rate Three Color Marker (srTCM) [4] measures the traffic stream and marks its packets (green, yellow or red) according to three traffic parameters:

- The Committed Information Rate (CIR) configures the maximum bandwidth that can be allocated to a packet that is flowing under normal line condition.
- The Committed Burst Size (CBS) determines how large traffic bursts can be before some of the traffic exceeds the rate limit.
- The Excess Burst Size (EBS) configures the maximum number of bytes allowed for incoming packets to burst above the CIR and still be marked with medium-high packet-loss priority (yellow).

The traffic is then marked as follows:

- Traffic within CIR that does not exceed CBS always conforms and is marked green;
- Traffic that exceeds CBS but not EBS is dropped or marked yellow;
- Traffic that exceeds EBS is marked with red (high packet-loss priority).

3.2 Exceed Action

Once the packet is classified as exceeding a particular rate limit, the switch:

- drops the packet;

  or

- marks the packet with yellow or red color and continue.

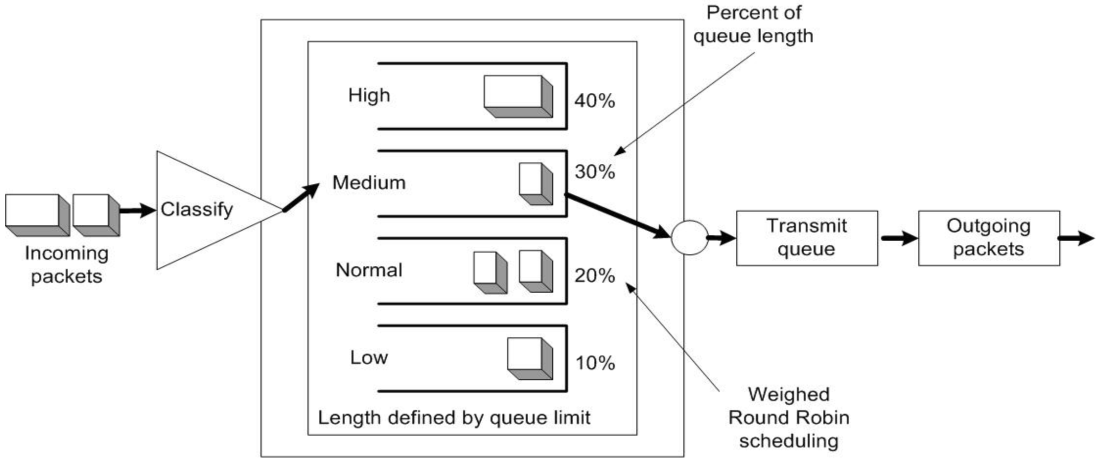

Fig. 3. Weighted Round Robin queuing

3.3 Color-Blind and Color-Aware

Rate limiting operates in one of the two modes:

- Color-Blind mode: assumes that the packet stream is uncolored;
- Color-Aware mode: assumes that the incoming packet stream was pre-colored, so each packet can be colored green, yellow or red.

4. NEW SCHEDULING COMMANDS

Two Hybrid scheduling commands were proposed in [8]. Four new Hybrid scheduling commands are implemented in this paper. Also, the output of show scheduling-profile command [8] was changed to display the new commands.

- scheduling-profile sp <*profile_number*> command [8] uses the Strict Priority (SP) scheduling.
  - *profile_number*: the scheduling profile ID. The range is <1–8>.
- scheduling-profile wrr <*profile_number*> <*q1*> <*q2*> <*q3*> <*q4*> <*q5*> <*q6*> <*q7*> <*q8*> command [8] applies and configures Weighted Round-Robin (WRR) scheduling.
  - <q1>…<q8>: the number of bytes assigned to each of the eight queues. The range is <1–255> bytes.
  - In WRR scheduling, bandwidth is allocated proportionally for each queue. Network resources are shared among all of the applications the user services, each having the specific bandwidth requirements that can be identified.
- scheduling-profile hybrid-1 <profile_number> <q1> <q2> <q3> <q4> <q5> <q6> <q7> command [8] applies and configures the first Hybrid QoS algorithm: q8 queue behaves according to SP scheduling, and the rest of the queues behave according to WRR scheduling.
- scheduling-profile hybrid-2 <profile_number> <q1> <q2> <q3> <q4> <q5> <q6> command [8] applies and configures the second Hybrid QoS algorithm: q7 and q8 queues behave according to SP scheduling, and the rest of the queues behave according to WRR scheduling.
- scheduling-profile hybrid-3 <profile_number> <q1> <q2> <q3> <q4> <q5> command applies and configures the third Hybrid QoS algorithm: q6, q7, and q8 queues behave according to SP scheduling, and the rest of the queues behave according to WRR scheduling.
- scheduling-profile hybrid-4 <profile_number> <q1> <q2> <q3> <q4> command applies and configures the fourth Hybrid QoS algorithm: q5, q6, q7, and q8 queues behave according to SP scheduling, and the rest of the queues behave according to WRR scheduling.
- scheduling-profile hybrid-5 <profile_number> <q1> <q2> <q3> command applies and configures the fifth Hybrid QoS algorithm: q4, q5, q6, q7, and q8 queues behave according to SP scheduling, and the rest of the queues behave according to WRR scheduling.
- scheduling-profile hybrid-6 <profile_number> <q1> <q2> command applies and configures the sixth Hybrid QoS algorithm: q3, q4, q5, q6, q7, and q8 queues behave according to SP scheduling, and the rest of the queues behave according to WRR scheduling.
- show scheduling-profile [<*profile_number*>] command displays the scheduling profile configuration for all profiles or for the specified profile ID.
  - *profile_number*: (optional) the scheduling profile ID, in the range of <1–8>. If the profile ID is not specified, all scheduling profiles are displayed.

5. TESTING RESULTS

The following configuration example is based on the new Hybrid algorithm commands and shows the results obtained after configuring the commands listed in the section above:

- Apply SP on all 8 queues:

(qos)#scheduling-profile sp ***7***

- Apply WRR on all 8 queues:

(qos)#scheduling-profile wrr ***8 2 1 2 3 4 1 0 7***

- Apply first Hybrid algorithm:

(qos)#scheduling-profile hybrid-1 ***1 2 1 2 3 4 1 2***

- Apply second Hybrid algorithm:

(qos)#scheduling-profile hybrid-2 ***2 2 1 2 1 2 1***

- Apply third Hybrid algorithm:

(qos)#scheduling-profile hybrid-3 ***3 2 1 2 1 1***

- Apply fourth Hybrid algorithm:

(qos)#scheduling-profile hybrid-4 ***4 1 1 2 4***

- Apply fifth Hybrid algorithm:

(qos)#scheduling-profile hybrid-5 ***5 2 3 2***

- Apply sixth Hybrid algorithm:

(qos)#scheduling-profile hybrid-6 ***6 2 2***

- Display the scheduling profile configuration:

#show scheduling-profile

Id	Type	Q 1	Q 2	Q 3	Q 4	Q 5	Q 6	Q 7	Q 8
7	sp	-	-	-	-	-	-	-	-
8	wrr	2	1	2	3	4	1	0	7
1	hybrid-1	2	1	2	3	4	1	2	-
2	hybrid-2	2	1	2	1	2	1	-	-
3	hybrid-3	2	1	2	1	1	-	-	-

4	hybrid-4	1	1	2	4	-	-	-	-
5	hybrid-5	2	3	3	-	-	-	-	-
6	hybrid-6	2	2	-	-	-	-	-	-

After applying all the commands, the following results are obtained (as the output of show scheduling-profile command displays):

- On profile with ID 7, all 8 queues are using SP scheduling.
- On profile with ID 8, all 8 queues are using WRR scheduling. The queues have now assigned different amount of bytes that can be transmitted.
- On profile with ID 1, queue Q8 behaves according to SP scheduling, and the rest of queues behave according to WRR scheduling. Queues Q1-Q7 have now assigned different amount of bytes that can be transmitted.
- On profile with ID 2, queues Q7 and Q8 behave according to SP scheduling, and the rest of queues behave according to WRR scheduling. Queues Q1-Q6 have now assigned different amount of bytes that can be transmitted.
- On profile with ID 3, queues Q6-Q8 behave according to SP scheduling, and the rest of queues behave according to WRR scheduling. Queues Q1-Q5 have now assigned different amount of bytes that can be transmitted.
- On profile with ID 4, queues Q5-Q8 behave according to SP scheduling, and the rest of queues behave according to WRR scheduling. Queues Q1-Q4 have now assigned different amount of bytes that can be transmitted.
- On profile with ID 5, queues Q4-Q8 behave according to SP scheduling, and the rest of queues behave according to WRR scheduling. Queues Q1-Q3 have now assigned different amount of bytes that can be transmitted.
- On profile with ID 6, queues Q3-Q8 behave according to SP scheduling, and the rest of queues behave according to WRR scheduling. Queues Q1 and Q2 have now assigned different amount of bytes that can be transmitted.

6. CONCLUDING REMARKS

Quality of Service (QoS) provides different priorities to different applications, users, or data flows, or guarantees a certain level of performance to a data flow. For example, a required bit rate, delay, jitter, packet dropping probability, and/or bit error rate may be guaranteed.

Requirements for different types of packet traffic are specified through the QoS constraints [3]:

- constraints that accounts (additive);
- constraints that are breeding (multiplicative);
- constraints that are selected according to the smallest or bigger value (concave or convex).

QoS routing finds a path from a source to a destination and satisfies certain specified constraints. QoS routing consists of two phases:

- collecting and updating/maintaining the information required for QoS routing process;
- searching for possible ways, based on information collected in the first phase.

Traffic Scheduling allows controlling the packets transmission - based on priorities assigned to packets - and selecting a queuing algorithm.

To improve the QoS scheduling, one of queuing algorithms can be applied: Strict Priority (SP), Weighted Round Robin (WRR), and Hybrid. The Hybrid algorithm, which combines SP and WRR scheduling, proposed in [8] was improved by adding four new commands. Now each queue can have assigned different amount of bytes that can be transmitted.

REFERENCES

[1] Campbell, A. and Coulson, G. (1997). A QoS adaptive multimedia transport system: design, implementation and experiences, *J. Distributed Systems Engineering*, vol. 4, no. 1, pp. 48-58.

[2] Chuah, C.N., Subramanian, L., Katz, R. H., and Joseph, A. D.. (2000). QoS provisioning using a clearing house architecture, *IEEE/IFIP International Workshop on Quality of Service (IWQoS)*, Pittsburgh, Pennsylvania, June 2000, pp. 115-124.

[3] Crawley, E.S., Nair, R., Rajagopalan, B., and Sandick, H. (1998). *A Framework for QoS-based Routing in the Internet*, RFC Editor, USA, Aug. 1998, pp. 4-14.

[4] Heinanen, J., Finland, T., and Guerin, R. (1999). *A Single Rate Three Color Marker*, Category: Informational, University of Pennsylvania, pp. 2-5, Sept. 1999.

[5] Karam M.J. and Tobagi, F.A. (2002). Analysis of delay and delay jitter of voice traffic in the internet," *The International Journal of Computer and Telecommunications Networking*, vol. 40, no. 6, pp. 711-726.

[6] Kuipers, F.A., Korkmaz, T., Krunz, M., and Van Mieghem, P. (2002). An Overview of Constraint-Based Path Selection Algorithms for QoS Routing, *IEEE Communications Magazine, Special Issue on IP-Oriented Quality of Service*, vol. 40, no. 12, pp. 50–55.

[7] Radulescu T. and Coanda, H.G. (2007). *QoS in IP multimedia networks*, Editura Albastra, Cluj-Napoca.

[8] Stanica R. and Petre, E. (2010). Control the Packets Transmission Using Quality of Service Protocol, *Proceedings of the 14th International Conference on System Theory and Control (Joint conference of SINTES14, SACCS10, SIMSIS14),* Sinaia, Romania, Oct. 17-19, 2010 pp. 535-540.

[9] http://www.batm.com/int/index/en/sdoc/type/111

[10] http://www.cisco.com/en/US/docs/internetworking/technology/handbook/QoS.html