Annales UMCS Informatica AI XII, 1 (2012) 47–56
DOI: 10.2478/v10065-012-0007-1

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI
http://www.annales.umcs.lublin.pl/

A method for QoS differentiation in DiffServ networks based on the long-term properties of a video stream

Zbigniew Omiotek[1]*

[1] *Department of Computer Science and Knowledge Engineering*
Zamość University of Management and Administration,
Akademicka 4, 22-400 Zamość, Poland

Abstract – This paper presents a method for adjusting the level of services offered by the network with quality of service differentiation for the long-term characteristics of a transmitted video stream. The Drop Precedence (DP) field located in the header of IP packet for this purpose was used.The DP field is set dynamically, based on the measurement of the long-term properties of a source video stream entering the network. The level of traffic perturbations present in a stream is expressed by the Hurst parameter, and then mapped to the size of a priority encoded in the DP field. By that means, an adaptive differentiation of the preferences of individual streams within the same AF PHB class of service is implemented, depending on the size of perturbations existing in the flow. The use of the long-term Hurst parameter, as a criterion of classification, makes the treatment of packets marked with a given priority value does the job well on a larger time scale.

1 Introduction

For some time we have observed rapid development of multimedia applications, which necessitates increasing quality of service (QoS) guarantee offered by a network. Therefore, an intensive study is performed, i.e. in the area of computer networks with the QoS differentiation (Differentiated Services, DiffServ). Their effects are, for example, the proposals for new adaptive algorithms for the network access control or the methods for class of service allocation in a dynamic way [**1, 2, 3, 4, 5, 6**]. However, a limitation of the proposed solutions is that they involve short-term nature of network processes, and do not include a traffic self-similarity [**7, 8, 9, 10**]. As a result, limited network resources may not be used in an efficient manner. The results of numerous

*zomiotek@gmail.com

studies confirm the presence of self-similarity in both compressed streams of frames of the video sequences, as well as streams of packets used for transmission of these sequences over a network [**11, 12, 13**]. Perturbations occurring in a video stream can not be eliminated because they are a natural feature of information contained in the source image. But we can try to adjust a transmission path to the video stream. To achieve this goal, the use of perturbations measure in the form of a long-term Hurst parameter [**14, 15**] in this paper was proposed. On this basis, the packets can be processed adaptively at each node along the transmission path. This task requires the information about how to service the stream to be transmitted in every packet of the stream. After reading this information, the nodes should be treated preferentially as for the packets belonging to the streams characterized by higher levels of perturbations.

Requirements for packets handling can be saved in the Differentiated Service Code Point (DSCP) located in the IP header (Fig. 1) [**16**]. In the case of IPv4, the DSCP field is contained in the Type of Service (TOS) field [**17**], and for IPv6 in the Traffic Class (TC) field [**18**]. In the context of differentiated services, both of these fields, together with two bits not usedare included in the Differentiated Services (DS) field.

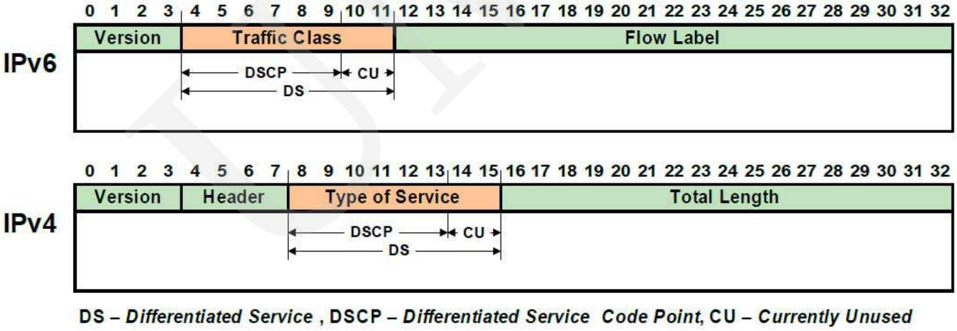

Fig. 1. DSCP field in the header of IPv6, and IPv4 packets.

Currently, the DSCP field is used by the routers capable of differentiating between the DiffServ traffic, and therefore the concept of improving the quality of service proposed in this paper can be seen as an extension of the DiffServ model [**19**].

The second section of this paper provides an overview of classes of services available in the DiffServ architecture. The concept of dynamic management of a priority flow based on the long-term characteristics of a source video stream is presented in section three. The fourth section discusses a key part of the system, which is a Priority Management Module. The last part of the paper presents the conclusions.

2 Classes of service in the DiffServ networks

Previous activities on the DiffServ architecture standardization led to define following classes of service: Best Effort (BE), Expedited Forwarding (EF) [**20**], and Assured Forwarding (AF) [**21**]. BE PHB (Per-Hop Behaviour) is an unclassified service, EF PHB is used to transfer the packets requiring a real-time service, and AF PHB group is responsible for packets transmission across multiple classes of an elastic traffic. For the AF PHB group 4 classes of service are defined, which correspond to the following values of the Class Selector (CS) in the DSCP field (Fig. 2):
– 001 (class 1);
– 010 (class 2);
– 011 (class 3);
– 100 (class 4 - the best service).

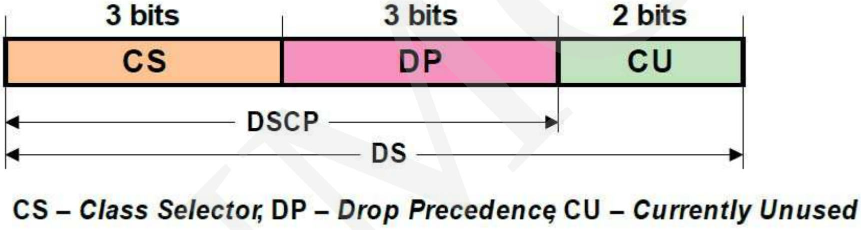
3 bits
3 bits
2 bits
CS
DP
CU
DSCP
DS
CS – *Class Selector*, DP – *Drop Precedence*, CU – *Currently Unused*

Fig. 2. Structure of the DS field.

Three groups differ in a packet drop probability when queue overflows within each defined class. The Drop Precedence (DP) field is used to encode the level of a packet drop. DP may take the following values:
– 110 (level 3, high - the least important priority);
– 100 (level 2, medium);
– 010 (level 1, low).

The recommended DSCP field values for each class of service within AF PHB are presented in RFC 2597 [**21**]. Each of these classes can be described as A_{ij}, where: $1 \leq i \leq 4$ is the number of a class of service, and $1 \leq j \leq 3$ is the probability level of a packet drop. For the EF PHB class DSCP equals the reserved code 101110, while for BE PHP it is equal to 000000. The AF PHB classes recommended for the use are shown in Table 1.

Currently, the CS and DP codes are usually set in an administrative way. However, the proposals for a dynamic class of service assignment do not include the long-term processes present in the transmitted streams. Such approach to the classification of flows makes that the network resources are not optimally used by various streams of packets. The CS field value follows directly from the Service Level Agreement (SLA) between the user, and the DiffServ provider, so a policy of manually setting a class of stream is difficult to change. But we can try to set the DP field values in an adaptive

Table 1. Recommended DSCP codes for different classes of service in AF PHB.

Class of	The level of packet drop		
service	Low	Medium	High
Class 1	001010	001100	001110
Class 2	010010	010100	010110
Class 3	011010	011100	011110
Class 4	100010	100100	100110

way. Then preferences differentiation of individual streams within the same class of service will depend on a size of the current perturbations in the flow.

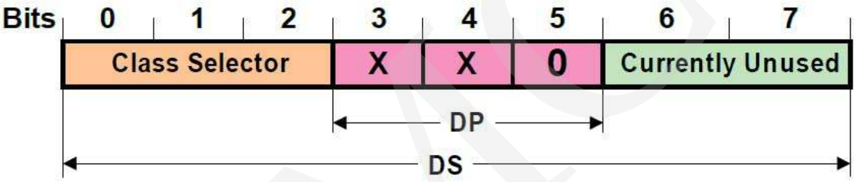

Fig. 3. Structure of the DP field.

Fig. 3 shows the structure of the DP field. This field consists of three bits, where bit number 5 (using the numbering for the entire DS field) has a fixed value of 0. The values of bits 3 and 4 are changed and they are used to encode the information about drop probability (priority) of the packet. Low value of the probability corresponds to the DP code of 01. Medium and high probabilities are coded as 10 and 11 respectively. This notation is common to all four AF classes. The assignment presented above is recommended by IETF [**21**]. But it is assumed that the domain administrator may use any links, as well as define other classes of service.

3 Concept of dynamic management of a priority flow

Normally, the operation of packets classification, and assigning to them a specific priority within the class of service are implemented in the boundary router of the DiffServ domain. However, the applications can perform packet marking with the DP code as well. If the application is considered by the domain to be trusted, then the packet assigning to a particular class of service will be respected in the domain. T he concept presented in this paper assumes the packets marking with the DP code by the streaming server software, which is recognized by the DiffServ domain as a trusted application.

Fig. 4 shows the idea of packet marking with the DP code by the source of a stream sent to the network. The figure presents a case for the image source introducing a

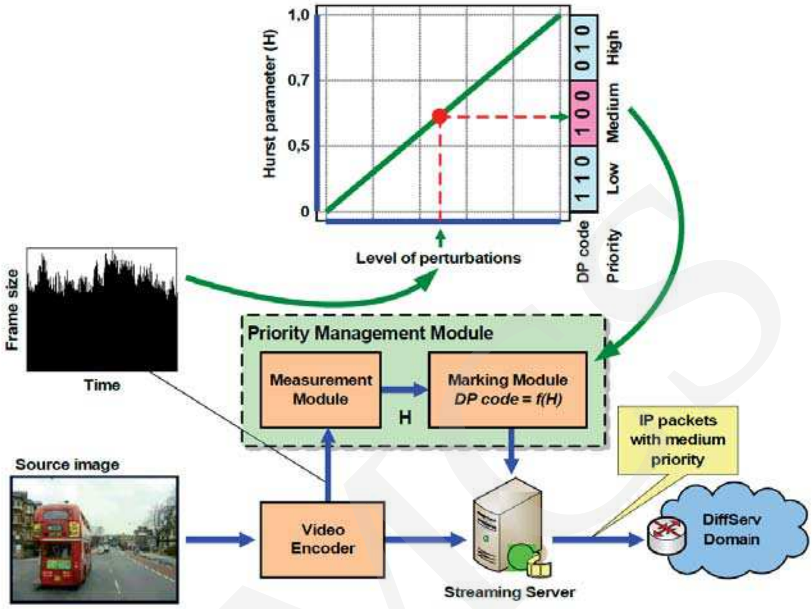

Fig. 4. Packets marking in terms of the medium perturbations level of a source stream.

medium level of perturbations. If this level changes, the packets will be marked with an appropriate priority (low, medium or high) according to the value of the measured Hurst parameter.

4 Priority Management Module

From the viewpoint of ensuring QoS, a crucial element of the system shown in Fig. 4, is the Priority Management Module (PMM). PMM, as a functional element of the streaming server software, is scheduled for implementation in the future research. Its main function will be measurement of the statistical properties of a stream of the compressed video frames, and decision how to mark the packets that are currently sent to the network. PMM will consist of two components: the Measurement Module, and the Marking Module.

4.1 Measurement Module

The Measurement Module will estimate the Hurst parameter based on the measurement of successive frames size of the compressed video stream. Two basic requirements should be fulfilled during the measurement process. In order to ensure a high accuracy of the H parameter estimation, the length of time series (the number of frames used for measurement) should be large enough. On the other hand, it must be ensured that the reaction time (inertia) to the process changes in the processed signal is the smallest. To meet these requirements, the measurement method using the sliding time window with the fixed width was selected. The above-mentioned window has two parameters: the width (n) which corresponds to a fixed number of frames used in the process of the H parameter estimation, and the shift (Δn) which is the number of frames by which a window will be moved to update the input data.

The idea of measurement using the sliding window is shown in Fig. 5. The measurement process begins when the time $t = t_0$. Then, n compressed frames of the stream are in the time window. This first measurement stage is the longest as a buffer that stores data needed to collect information on the n frames. When the buffer is filled, the value of the Hurst parameter (H_1) is estimated, and on this basis, the packets priority is established. The packets sent at this time by the streaming server (T_1) are marked with the default code ($DP_0 = 100$), corresponding to the medium priority. The time that elapses from the moment of collecting the data for n frames, until emergence of an important priority at the output of the Marking Module, is defined as a processing time (t_p). It includes the execution time for memory operation, and the execution time of algorithms by the Measurement Module (estimate the value of H) and the Marking Module (determine the value of DP). Taking into account the basic performance parameters of the modern RAM (clock frequency, access time), it seems that a processing time will be negligibly small in comparison to the update time of the data in the measurement buffer.

The Measurement Module operates according to the algorithm shown in Fig. 6. The time window moves by Δn frames to the right on a timeline at the time $t = t_0$. Then the information about the size of Δn the oldest frames is removed from the buffer, and the other content is moved according to the *FIFO* algorithm. Then the youngest items in the buffer are filled with the data about the size of Δn new current frames. Thus, the information about the size of a fixed number of frames equal to n is all time in the buffer. The value of the Hurst parameter estimated in this period of time by the Measurement Module is H_2. The packets containing the frames currently sent to the network (T_2) are marked with the DP_1 priority. It corresponds to H_1 parameter estimated in the previous measurement period (T_1). The time window shifts by Δn consecutive frames at the time $t = t_2$. The packets currently sent to the network (T_3) are marked with the DP_2 code corresponding to H_2. At the same time, the value of H_3 for the packets, which will be sent in the next period of time (T_4) is estimated. The whole measurement process is repeated over the time periodically.

4.2 Marking Modul

The result returned by the Measurement Module will be an input parameter for the Marking Module. It will operate as a decision-making unit. It will define a way of the packets marking with the DP code during the current period of time T, based on the Hurst parameter value estimated in the previous measuring period T_{n-1}. As noted

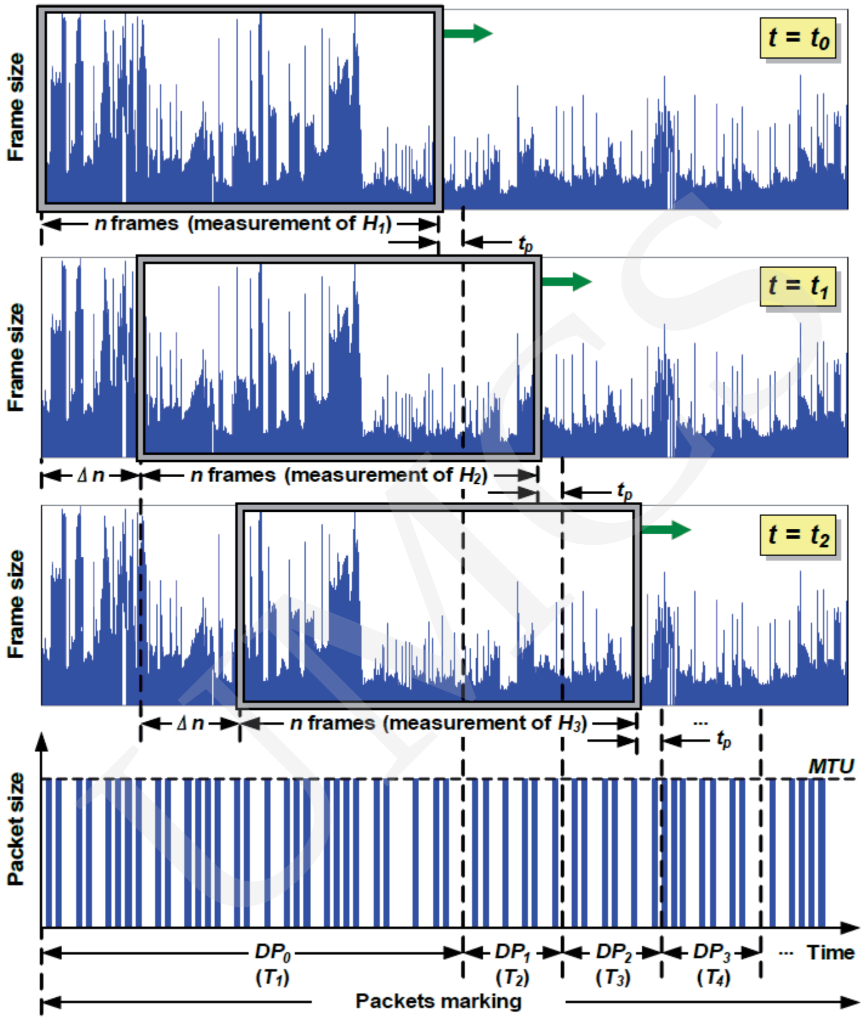

Fig. 5. Idea of measuring and packets marking by the Priority Management Module (t_p – processing time, *MTU* – *Maximum Transmission Unit*).

earlier, in the first measurement period (T_1) the packets will be marked with a default code ($DP_0 = 100$), corresponding to the medium priority. The following pseudocode describes how the Module Marking works:

```
  if (hurst > 0.7) then DP = 010 else if (hurst > 0.5) and (hurst <=
0.7) then DP = 100 else DP = 110 endif
```

Within each class of service there are three standard priorities. Therefore, for the Hurst parameter there have been selected two thresholds, which correspond to the

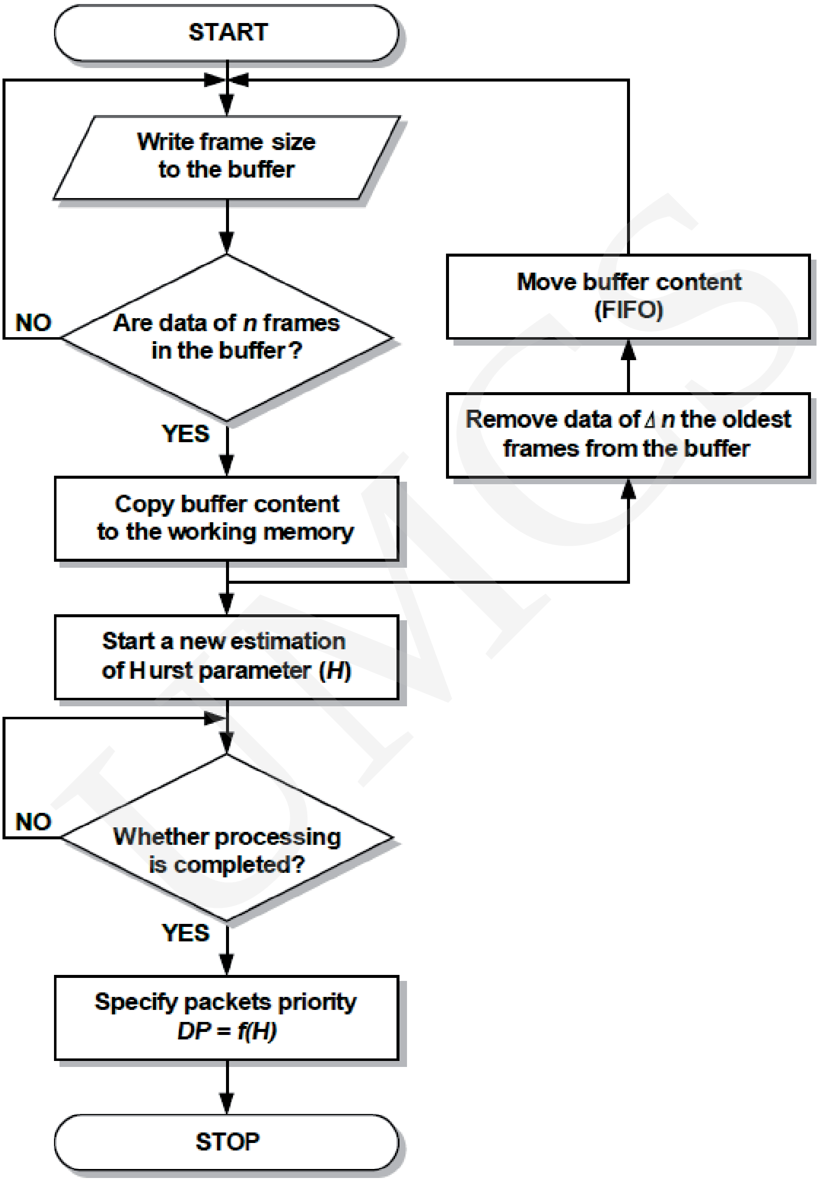
START
Write frame size to the buffer
Are data of *n* frames in the buffer?
NO
YES
Copy buffer content to the working memory
Move buffer content (FIFO)
Remove data of Δ *n* the oldest frames from the buffer
Start a new estimation of Hurst parameter (*H*)
Whether processing is completed?
NO
YES
Specify packets priority *DP = f(H)*
STOP

Fig. 6. Priority Management Module algorithm.

specified packet drop probability. However, these thresholds will be verified by the experiment. If the processes introduced by a source of image result in the large flow perturbations ($H > 0.7$), then the packets are marked with a high priority in the next measurement period ($DP = 010$), and the drop probability of the packets waiting for service in the network nodes is the smallest. When the perturbations introduced by a source are smaller ($0.5 < H <= 0.7$), then the packets are marked with a medium priority ($DP = 100$). All the other cases, satisfying the condition $H <= 0.5$ are marked with the lowest priority.

The Marking Module will be closely integrated with the network layer software of a streaming server. This will make an automatic online DP code modification possible in the DSCP field of the IP packet. The Marking Module may also specify how to mark the TC field. Then the code for this field would depend on SLA agreement between the user and the DiffServ domain provider.

5 Conclusions

There is a fundamental difference between the existing methods of QoS provided in the DiffServ architecture, and the method proposed in this paper. In the classic DiffServ model, stream assignment to the class of service, and setting the preferences within the class done by hand and this is a static operation. As a result, the packets with a lower drop probability can always be given a preferential treatment, regardless of the actual needs. Although new approaches introduce the concepts of a dynamic classification of the input streams in the classification process, they use the short-term parameters as the control variables. Both methods lead to an inefficient use of the limited network resources. In fact, the network traffic is self-similar, and its processes are characterized by the long-term dependencies. This fact should be taken into account when designing new mechanisms to improve QoS guarantees in the DiffServ networks. In accordance with the proposed idea, the packets preferences will be set in a dynamic way, according to the current level of perturbations existing in a flow. The size of these perturbations is closely related to the level of a stream self-similarity. Consequently, the packets belonging to the same flow will have different preferences depending on the actual needs.

References

[1] Bay Y., Class-Based Packet Scheduling to Improve QoS for IP Video, Telecommunication Systems 29(1) (2005): 47.

[2] Boucadair M., Levis P., Griffin D., Wang N., Howarth M., Pavlou G., Mykoniati E., Georgatsos P., Quoitin B., Quoitin B., Rodriguez Sánchez J., Garcia-Osma M. L., A Framework for End-to-End Service Differentiation: Network Planes and Parallel Internets, IEEE Communications Magazine 45(9) (2007): 134.

[3] Ahmed T., Boutaba R., Mehaoua A., A measurement-based approach for dynamic QoS adaptation in DiffServ networks, Computer Communications 28 (2005): 2020.

[4] McAllister B., Marshall A. J., Woods R. F., A programmable architecture for layered multimedia streams in IPv6 networks, Journal of Networks 5(1) (2010): 65.

[5] Ahmed T., Mehaoua A., Boutaba R., Iraqi Y., Adaptive Packet Video Streaming Over IP Networks: A Cross-Layer Approach, IEEE Journal on Selected Areas in Communications 23(2) (2005): 385.

[6] Chen L., Liu G., Zhao F., An Improved Marking Mechanism for Real-Time Video over DiffServ Networks, Lecture Notes in Computer Science 4810 (2007): 510.

[7] Willinger W., Leland W. E., Taqqu M. S., Wilson D. V., On the Self-Similar Nature of Ethernet Traffic (extended version), IEEE/ACM Transactions on Networking (1994): 1.

[8] Crovella M. E., Bestavros A., Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes, IEEE/ACM Transactions on Networking 5(6) (1997): 835.

[9] Park K., Kim G., Crovella M., On the effect of traffic self-similarity on network performance, Technical Report CSD-TR-97-024, Purdue University, Dept. of Computer Sciences (1997).

[10] Csabai I., 1/f Noise in Computer Network Traffic, Journal of Physics A A27 (1994): 417.

[11] Garrett M. W., Willinger W., Analysis, modeling and generation of self-similar VBR video traffic, Proc. Sigcomm '94, September (1994): 269.

[12] Krunz M., Sass R., Hughes H., Statistical characteristics and multiplexing of MPEG streams, Proc. of IEEE INFOCOM (1995): 455.

[13] Hong S. H., Park R.-H., Lee C. B., Hurst Parameter Estimation of Long-Range Dependent VBR MPEG Video Traffic in ATM Networks, Journal of Visual Communication and Image Representation 12 (2001): 44.

[14] Mandelbrot B. B., Van Ness J. W., Fractional Brownian motion, fractional noises and applications, SIAM Review 10 (1968): 422.

[15] Bassingthwaighte J. S., Raymond G. M., Evaluating rescaled range analysis for time series, Ann. Biomed. Engin. 22 (1994): 432.

[16] Nichols K., Blake S., Baker F., Black D., Definition of the Differentiated Services Field (DS Field) in the Ipv4 and Ipv6 Headers, IETF RFC 2474 (1998).

[17] RFC 791, Internet Protocol (IP) Specifications (1981).

[18] Deering S., Inden R., Internet Protocol, Version 6 (IPv6) Specification, IETF RFC 2460 (1998).

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

[20] Jacobson V., Nichols K., Poduri K., An Expedited Forwarding PHB, IETF RFC 2598 (1999).

[21] Heinanen J., Baker F., Weiss W., Wroclawski J., Assured Forwarding PHB Group, IETF RFC 2597 (1999).