CONTRIBUTION TO ADAPTIVE SAMPLING OF QoS PARAMETERS IN COMPUTER NETWORKS[1]

*Juraj GIERTL, *František JAKAB, *Ján BAČA, **Rudolf ANDOGA, ***Michal MIRILOVIČ
*Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic, tel.: + 421 55 602 2553, E-mail: Juraj.Giertl@tuke.sk
**Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic tel.: + 421 55 602 4213, E-mail: Rudolf.Andoga@tuke.sk
***Department of Electronics and Multimedia Communications, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic, tel.: + 421 55 602 2853, E-mail: Michal.Mirilovic@tuke.sk

SUMMARY

This paper focuses on adaptive sampling that is used by measurement and evaluation of operational parameters of computer networks. These parameters characterize networks' ability to maintain quality of services (QoS) defined by volume and time operational characteristics. Process of measurement can be performed using several methods. From the measurement efficiency point of view, the most appropriate are the so-called passive methods. No testing traffic needs to be generated while using them, and parameter values are derived from ongoing network activity. This area's forming standard that has appeared recently is referred to as the standard for exporting information on IP flows (IPFIX).

However, there are several problems related to measurement of network's QoS parameters. The most important ones include a huge amount of data that needs to be processed on the measurement platform and afterwards to be stored for potential future use. This issue has become critical as the speed of currently used network technologies increased. However, there are several methods of reducing the amount of processed data. One of them is sampling which makes possible to select a characteristic sample of the overall packet population. Recent sampling methods are inefficient because they do not reflect on the nature of ongoing network activity. This problem, though, can be overcome by the so-called selective sampling. The paper intends to outline a model for adaptive sampling based on modification of the sampling interval by fuzzy logic controller. Experiments leading to the optimization of its parameters are also presented. At the end, the proposed methodology is compared to conventional sampling methods.

Keywords: *QoS, IPFIX, computer network, adaptive sampling, fuzzy logic controller*

1. INDRODUCTION

Measurement and evaluation of computer networks' operational parameters are related to various services they provide. For example, by using the data obtained from the network monitoring, it is possible to verify the fulfillment of contractual obligations between customers and Internet service providers. Providers can also use the data for creating a usage-based billing or detecting intrusions threatening the network's security.

Specific time characteristics (e.g., one way delay, jitter) are important for applications working with real-time data and requiring a quick response from the network – mainly sound and video transmissions [10] [11]. As a result, IP telephoning has been recently introduced as an effective worldwide communication tool. In addition, because of introduction of high quality network infrastructures and technologies with high data throughput, videoconferences and IP streaming are gradually becoming more popular communication tools.

Using passive methods for measurement and evaluation of high-speed network operational parameters within ongoing network activity might cause several problems:

- Huge amount of data that needs to be stored,
- High time demands for processing and evaluation of obtained values.

The problem's solution lays in reducing the amount of data that needs to be processed and evaluated while at the same time the required measurement accuracy must be respected. It is possible to use either several optimization methods [4], or their combinations. Volume reductions are achieved by application of a sampling method that will generate a packet sample representing sufficiently the whole packet population. Another option is to either filter out all packets irrelevant to the measurement process [14], [15], or to select and process only a fragment of a packet (i.e., its head). On its basis, it is then possible to aggregate packets into flows in accordance with the recent IPFIX standard (IP flow information export) [12]. By using this data procession, it is possible to export smaller data volumes from measurement points designed for additional analysis. In order to define an aggregation, it is possible to use templates typical for a given protocol, such as IPFIX [2], NetFlow [1].

The following sections discuss the sampling of network activity and specific fuzzy logic based adaptation methods.

2. INTRODUCTION TO SAMPLING

Sampling is focused on selecting the most representative sample of a network activity. It is used for obtaining information on the whole group

[1] *This work is partially supported by the Slovak Science Grant Agency (VEGA No 1/2175/05 "Evaluation of operational parameters in broadband communication infrastructures: research of supporting platforms").*

of monitored packets, without the obligation of processing them all. The selection can depend on the position of the packet, packet content and/or pseudo-random decisions.

Application of a sampling method, aims to gather information about a specific feature of the whole packet population. Therefore, the analysis of a sampled population is considerably less complex compared to analysis of the whole population. In order to obtain the required information with certain degree of accuracy, it is necessary to plan an appropriate sampling strategy. The main difference between sampling and filtering lays in nondeterministic selection that does not exclusively depend on packet's content.

The sampling intensity is indirectly proportional to the length of time interval between the selection of the two neighboring samples of a given network activity.

Small intervals will create a relatively precise view of the network's activity. However, in high-speed networks, intensive sampling can overload system devices. To solve this problem, one might apply larger sampling intervals, which in turn reduce the measurement accuracy.

Ordinary sampling methods are based on sampling intensity that is independent on intensity and fluctuations of network activity. They include:

- systematic sampling based on time or packets' count,
- random sampling with uniform or non-uniform probability distribution,
- stratified random sampling, also referred to as n-of-N.

Basically two situations might appear when using these methods:

- if the activity fluctuation is high a risk of loosing important information on network activity appears as a result of large sampling intervals,
- if the activity fluctuation is small, system devices are used inefficiently due to unnecessarily small sampling interval.

Presented drawbacks might be removed by adaptive sampling, which modifies the sampling intensity depending upon the nature of network activity. Provided that the fluctuation is high, the sampling intensity increases, if it is low, the intensity decreases. As a result, the use of system devices will get more efficient and smaller data volumes will be sent for procession.

Adaptive sampling can be also based upon the linear prediction, fuzzy logic or neural networks. Every method differs from each other, by the method used for regulation of sampling intervals. The following section discusses a method that uses fuzzy logic. The implemented method improves the method presented in [5].

3. SAMPLING BASED ON FUZZY LOGIC CONTROLLER

The use of fuzzy logic is appropriate in non-linear systems. Especially in those that are hardly to model or their exact modeling is completely impossible. All methods based upon fuzzy logic simulate human thinking by applying a set of rules based on one or multiple assumptions and simple implications. For example, if the following assumptions are valid "network load is unchanged" and "the sampling interval is small", the following implication might be used "significantly increase the sampling interval".

Figure 1 shows the concept of adaptive sampling based on fuzzy logic.

Using packets' information caught by a network interface card of a measurement station the value of monitored parameter of the network activity is calculated. It is obvious that this parameter changes over time and, therefore, is characterized by a sequence of values in time frames of a predefined size. Most often, the most interesting value of a given parameter is calculated over the time frame of 1 second (e.g., bandwidth in bytes per second). However, in order to obtain more precise results, it is recommended to decrease the time interval. The resulting parameter value is then calculated upon the time frame according to user requirements.

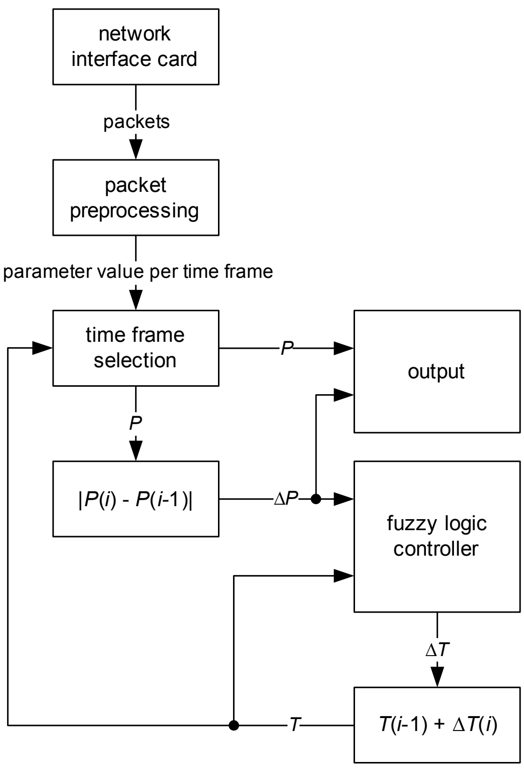

Fig. 1 Concept of adaptive sampling based on fuzzy logic

The key feature of adaptive sampling concept based on fuzzy logic is the fuzzy logic controller (FLC) that has two inputs and one output. The first input is a change of the size in a given parameter defined by the absolute value of difference between parameter values selected in two consecutive iterations (1).

$$\Delta P(i) = |P(i) - P(i-1)| \tag{1}$$

The second input is the size of a sampling interval defined by the formula (2):

$$T(i) = T(i-1) + \Delta T(i) \tag{2}$$

In which $T(i-1)$ is the size of the sampling interval in previous iteration and $\Delta T(i)$ is the change in size of the sampling interval defined by the FLC output of the previous iteration.

FLC output is defined as follows:

$$\Delta T(i+1) = f[\Delta P(i), T(i)] \tag{3}$$

Depending on the size of the sampling interval defined by the formula (2), appropriate time frames are selected. The values of monitored parameter within selected frames are used in the following iteration in formula (1) and then exported for procession in other parts of the tool. In case $\Delta P(i-1) = 0 \wedge \Delta P(i) = 0$ the export is unnecessary.

The function f in formula (3) cannot be expressed explicitly because it is affected by the FLC behavior. FLC properties affecting its behavior (membership functions, rules and defuzzification process) are discussed in the following subsections.

3.1. FLC membership functions

FLC distinguishes several levels of input and output variables. Their real values must be therefore mapped to particular fuzzy levels. Functions presented in Fig. 2 are used for this specific purpose. The horizontal axis contains variables' real values and the vertical axis determines the degree of membership to particular fuzzy levels. The list of fuzzy levels of input and output variables is presented in Table 1 and Table 2, respectively.

Membership functions in Figure 2 have real values of input and output variables defined via parameters π, τ and δ. Their values will be determined in section 4.2. As for their meaning, they refer to the following:

- π – maximum size change of monitored parameter – depends on input data,
- τ – maximum size of sampling interval – depends on required data reduction rate by a user, or sampling accuracy,
- δ – maximum change in the size of a sampling interval – it affects the speed of adaptation of sampling interval.

3.2. FLC rules

FLC rules (Tab. 3) define the level of output variable for each combination of input variables. Each rule consists of two assumptions (i.e., levels of input variables) whose relationship is defined by AND logic operator and implication determining the level of output parameter. For example rule No. 1 means that if the value of monitored parameter remains unchanged and the sampling interval is small, increase significantly the sampling interval.

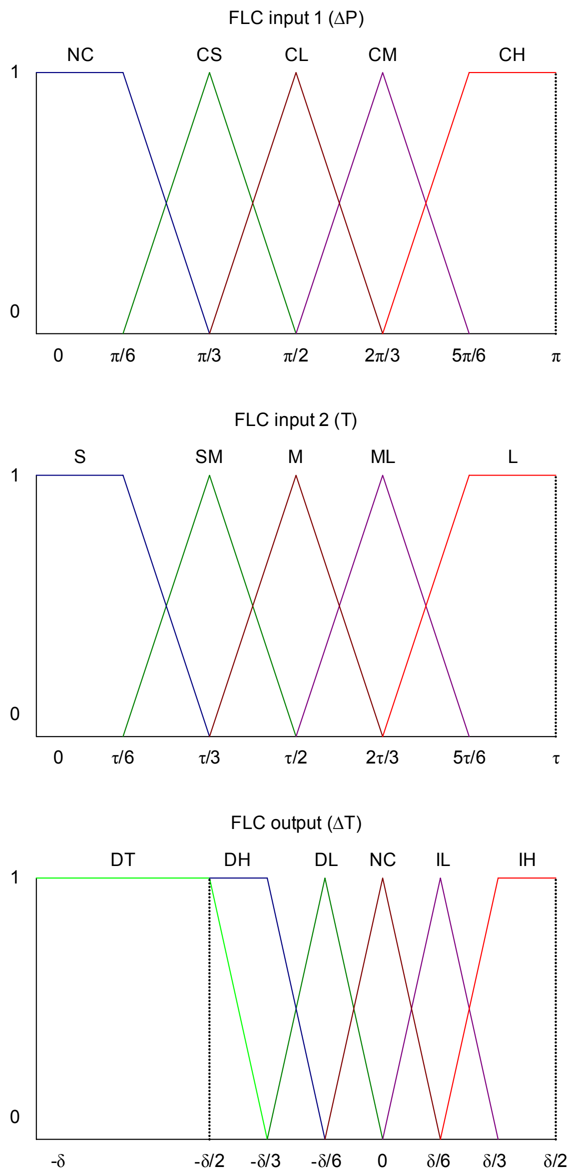

Fig. 2 FLC membership functions

ΔP		**T**	
NC	no change	S	small
CS	change slight	SM	small-medium
CL	change low	M	medium
CM	change medium	ML	medium-large
CH	change high	L	large

Tab. 1 Fuzzy levels of FLC input variables

ΔT	
DT	decrease tremendous
DH	decrease high
DL	decrease low
NC	no change
IL	increase low
IH	increase high

Tab. 2 Fuzzy levels of FLC output variable

Rule	ΔP	T	ΔT
1	NC	S	IH
2	NC	SM	IH
3	NC	M	IL
4	NC	ML	NC
5	NC	L	DL
6	CS	S	IH
7	CS	SM	IL
8	CS	M	NC
9	CS	ML	DL
10	CS	L	DL
11	CL	S	IL
12	CL	SM	NC
13	CL	M	DL
14	CL	ML	DH
15	CL	L	DH
16	CM	S	DL
17	CM	SM	DL
18	CM	M	DH
19	CM	ML	DH
20	CM	L	DH
21	CH	S	DL
22	CH	SM	DH
23	CH	M	DH
24	CH	ML	DT
25	CH	L	DT

Tab. 3 FLC rules

3.3. Deffuzification process

If the value of each input variable following the fuzzification process is assigned always to one membership function only one rule will be carried out (fired) and therefore there is no problem with determining the FLC's output. In general however, this is not the case. We often come to a case where after fuzzification, the input variable is defined by several fuzzy layers. It means that more than one rule will be applied and therefore identifying clear FLC output is problematic. This problem is solved in the process of deffuzification.

In our case, considering the membership functions as described in Fig. 1, and the rule base design (Fig. 3), it is only possible that the value of all input variables will be defined by membership degrees of two different fuzzy levels. In other words, only four rules at maximum will be applied. Although there are several methods of definition of FLC's output, in this case the following defuzzification process will be used:

1. Identify all rules to be applied with regard to the membership of inputs to fuzzy levels.
2. Select the minimum value of input variables' membership degree for each rule in step 1.
3. Use the α-cut of the output variable's membership function for each rule by the value selected in step 2.
4. Calculate the value of FLC output from α-cuts obtained in step 3 using the formula:

$$\Delta T = \frac{\sum_{j=1}^{n} x_j . y_j}{\sum_{j=1}^{n} y_j} \tag{4}$$

In which n is the number of applied rules, y_j is the value selected in step 2 for j-selected rule and x_j is the coordinate of the center on horizontal axis of the output variable's membership function for j-selected rule.

4. EXPERIMENTS

4.1. Experiment methodology

Experiments were carried out using Matlab Simulink with the Fuzzy logic toolbox. All scripts for data processing, FLC model and FLC based simulation model for adaptive sampling were designed using these tools.

Experimental data were obtained from the real TCP/IP activity of the network interface that provides connection of local network to the Internet. They were represented by the following sequence of pairs.

<time, size>

in which *time* is the time of a packet's arrival measured with the accuracy of 10^{-6} seconds and *size* refers to the packet's size in bytes. For the presentation purposes of this approach, a population of 12 934 packets recorded during 31 minutes was used.

This input sequence was used for creation of a sequence of values that would represent the network activity parameter (in this case the utilization of bandwidth) fixed to the time frame of 0,1 seconds. The newly created sequence was used as input information for sampling method testing. The concept of the experiment's implementation can be described in five steps:

1. To use a given sampling method for selection of a representative data sample. In real systems, they would be then exported for further processing in other parts of a measurement platform, where steps 2 and 3 are carried out.
2. To interpolate the missing samples using linear interpolation.
3. To recalculate sampled and referential data to the time frame of one second. By doing this, one can obtain the final data sequence which can be presented to a user in user-friendly format (e.g., graphical format – variable over a specified time period). The following two steps are necessary only for verification of different methods' accuracy. Therefore, they won't be used in a real system.
4. To modify sampled and referential data with the Myryl-Jege data transformation (Appendix A). This step, however, would not be necessary if we

compared only methods based on sampling time frame selection. Provided that we use methods based on packet selection, the resulting data will have different mean value and, therefore, the calculated variance of the referential signal will not be comparable to other methods. The main advantage of using the transformation is that the time dependency of a given variable is to be compared. However, this comparison will not represent the absolute measurement error.

5. To calculate the variance of sampled and referential data according to the following formula

$$error = \sum_{j=1}^{n} \left[p_{ref}\left(t_j\right) - p_{sampled}\left(t_j\right) \right]^2 \qquad (5)$$

In which n is the number of seconds of the input data sequence (in this case 1860), the sequence p_{ref} represents the referential parameter and the sequence $p_{sampled}$ represents the sampled parameter (both sequences are modified in previous steps).

4.2. Defining membership function parameters

The main drawback of adaptive methods is that the size of the final sample (the number of selected time frames of a sampled parameter) cannot be precisely set according to users' requirements. The sample size is affected by the size of the sampling interval, which does not remain constant in case of adaptive sampling (its values ranges from 1 to τ). By changing the parameter τ, it is possible to regulate the sample size. However it is not a continuous regulation rather a regulation in certain intervals.

During the experiments, different values of parameter τ were used, which in turn produced samples of different sizes. Parameters π and δ were optimized for each value of the parameter τ (Table 4). The optimization process was carried out with the emphasis on minimization of variance in the final sampled data compared to referential data. Parameters τ and δ are presented in units of selected time frame (in this case 0,1 seconds), parameter π is presented in units of monitored parameter (in this case bytes).

τ	π	δ	*sample size*	*error* [10^{-8}]
3	0,3.max(P)	1	50,6%	14276
4	0,3.max(P)	3	40,5%	27743
5	0,4.max(P)	3	33,7%	42353
6	0,4.max(P)	5	25,4%	67633
7	0,3.max(P)	4	20,6%	82804
8	0,5.max(P)	7	18,5%	143450
9	0,4.max(P)	8	17,0%	168820
10	0,4.max(P)	9	15,7%	181730
11	0,4.max(P)	4	14,6%	180950
12	0,4.max(P)	9	12,7%	269300

Tab. 4 FLC membership functions parameters

4.3. Comparison of sampling methods

Adaptive sampling based on FLC was compared to random sampling and stratified random sampling (sampling type of n of N) methods. Description of these methods is provided in [6].

Accuracy of particular methods is comparable only if they select samples of approximately same size. The comparison of variance from referential data for all cases, is presented in Table 4 and depicted in Figure 3.

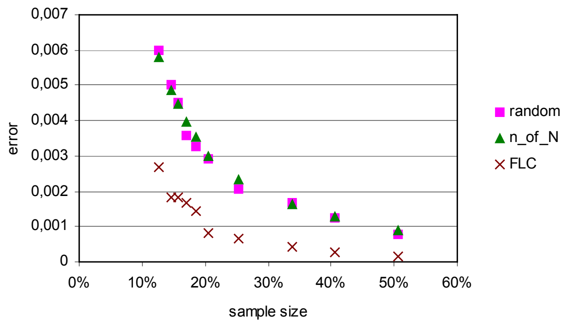

Fig. 3 Comparison of sampling methods

Examples of sampling method outputs with the sample size of approximately 20% are presented in Figures 5, 6 and 7. Referential data are shown in Figure 4.

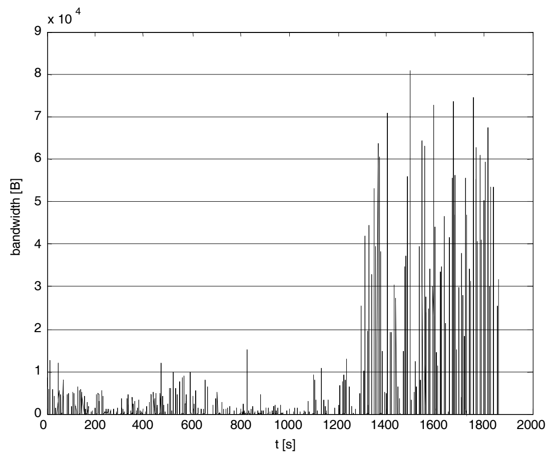

Fig. 4 Referential data

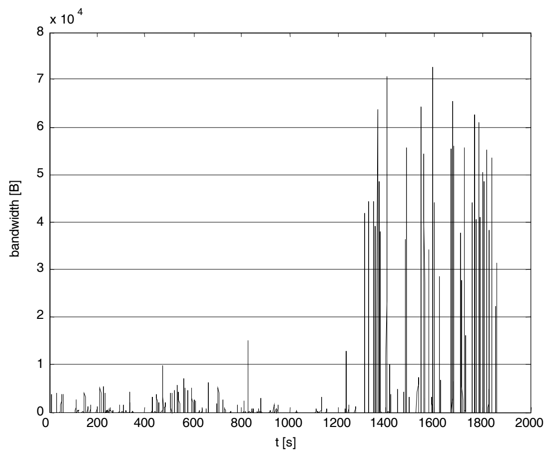

Fig. 5 Data sampled by FLC

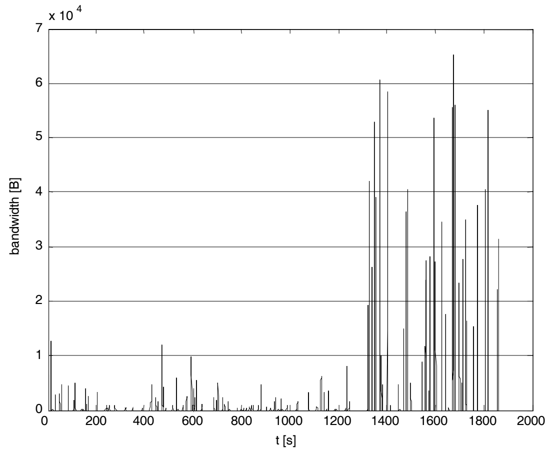

Fig. 6 Data sampled using random sampling

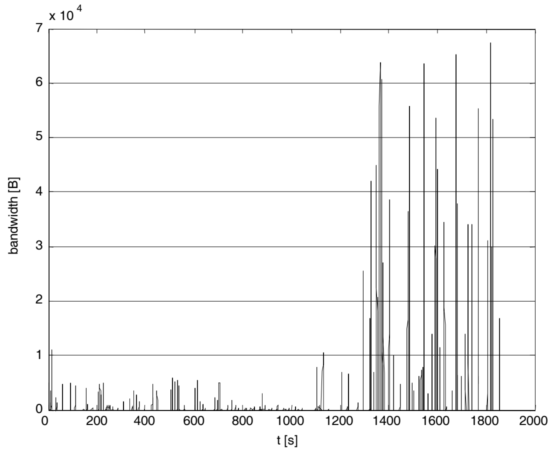

Fig. 7 Data sampled by random stratified sampling

5. CONCLUSION

Experimental results proved that adaptive sampling achieves better results than conventional sampling methods, ignoring the nature of network activity. Development of adaptive sampling methods based on FLC needs to be continued by solution of some other questions that might eventually improve the results of sampling:

1. Selection of a suitable time frame that would adequately represent a given parameter of the network activity and its potential adaptation with regard to the nature of ongoing activity, might significantly improve results compared to the methods based upon packet selection. However, the presented approach is less accurate compared to those methods (experiments with these methods are above the limits of this paper). On one hand, decreasing time intervals increases the sampling accuracy. Though, on the other hand if the value is too small the information contained in input variables required for changing the sampling interval gets lost. Data modified to a particular time frame must be sufficiently representative in order to depict the nature of trends by a certain network activity.
2. The FLC parameters presented in section 4.2 are suitable for sampling a random TCP/IP activity typical for Internet. Their modifications, however, can be used in case of other network technologies (ATM, MPLS).
3. The π parameter, which defines the sensibility of one FLC input, depends on the maximum value of the sampling parameter (Table. 4). Fixing this parameter during the sampling according to the changing nature of the activity will induce additional improvements in sampling results. For example, in case of activity represented by the data in Fig. 4 the overall sampling accuracy might be increased by using a different value of the parameter in the interval 0 to 1300 seconds and different value in interval 1300 - 1860 seconds.

Presented approach of adaptive sampling of network traffic satisfies the function of modifying the sampling intervals according to the nature of the network activity. Nevertheless, the problems outlined in previous sections require that the functions of this approach must be modified. This obstacle can be overcome by expanding the approach and include another adaptive mechanism that should optimize the approach during the sampling procedure. Methods described in [13] or self organizing neural networks seem to be the most suitable tool for this purpose. They are able to perform cluster analysis and upon the basis of its results they can modify FLC's parameters [3].

After successful completion, the presented approach will be implemented into the measurement tool "BasicMeter" [7] that is being developed within Computer network laboratories at the Department of computers and informatics of the Technical University in Kosice. This tool is designed in conformance with the IPFIX architecture, presented methods with other improvements [8] [9] will contribute to increasing the measurement efficiency.

6. APPENDIX A: MYRYL – JEGE DATA TRANSFORMATION

Definition:

Suppose we have two data sequences and each of them consists of n parameter values of the network activity fixed to a given time frame:

$$a(t) = f(t) \tag{6}$$

$$b(t) = \alpha g(t) + \beta \tag{7}$$

For which in case that $\forall j = 1,2,\ldots,n$ holds:

$$a(t_j) \geq 0\,,\ b(t_j) \geq 0 \tag{8}$$

and at the same time $\exists j$ those, that holds the following:

$$a(t_j) > 0\,,\ b(t_j) > 0 \tag{9}$$

Suppose that the following is true:

$$a'(t)=a(t)-\frac{1}{n}\sum_{j=1}^{n}a(t_j),\ b'(t)=b(t)-\frac{1}{n}\sum_{j=1}^{n}b(t_j) \quad (10)$$

$$a''(t)=\frac{a'(t)}{\sum_{j=1}^{n}|a'(t_j)|},\ b''(t)=\frac{b'(t)}{\sum_{j=1}^{n}|b'(t_j)|} \quad (11)$$

Formulas (10) and (11) are together called the Myryl – Jege transformation.

Quotation:

If the in the Myryl – Jege transformation holds that:

$$f(t)=g(t) \quad (12)$$

then:

$$a''(t)=b''(t) \quad (13)$$

Proof:

Incorporating formula (12) into the formula (7) and then applying equations (6) and (7) into formula (10) we have:

$$a'(t)=f(t)-\frac{1}{n}\sum_{j=1}^{n}f(t_j) \quad (14)$$

$$\begin{aligned} b'(t)&=\alpha f(t)+\beta-\frac{1}{n}\sum_{j=1}^{n}\left[\alpha f(t_j)+\beta\right]= \\ &=\alpha f(t)+\beta-\frac{1}{n}\left[n\beta+\sum_{j=1}^{n}\alpha f(t_j)\right]= \\ &=\alpha f(t)-\frac{1}{n}\sum_{j=1}^{n}\alpha f(t_j)= \\ &=\alpha\left[f(t)-\frac{1}{n}\sum_{j=1}^{n}f(t_j)\right]=\alpha a'(t) \end{aligned} \quad (15)$$

Applying the formula (15) into (11) we get:

$$b''(t)=\frac{\alpha a'(t)}{\sum_{j=1}^{n}|\alpha a'(t_j)|}=\frac{a'(t)}{\sum_{j=1}^{n}|a'(t_j)|}=a''(t) \quad (16)$$

REFERENCES

[1] Claise, B.: Cisco Systems NetFlow Services Export Version 9, RFC3954, October 2004.

[2] Claise, B., et. al.: IPFIX Protocol Specification, Internet Draft, IPFIX Working Group, September 2005.

[3] Czogaa, E., Czogala, E. and Leski, J.: Fuzzy and Neuro-Fuzzy Intelligent Systems, Springer, May 2000, pp. 209, ISBN 3790812897.

[4] Giertl, J.: Optimization Techniques for Measurement of QoS Parameters, In: 5th PhD Student Conference and Scientific and Technical Competition of Students of Faculty of Electrical Engineering and Informatics Technical university of Košice, Košice, Slovak Republic, 25 May 2005, pp. 45-46, ISBN 80-969224-4-0.

[5] Hernandez, E., Chidester, M. and George, A.: Adaptive Sampling for Network Management, Journal of Network and Systems Management, Vol. 9, No. 4, December 2001, pp. 409-434.

[6] Hronský, M., Jakab, F., Potocký, M., Jakab, R. and Giertl, J.: Sampling Algorithms for Nonintrusive Measurement in Network Oriented Educational Systems, In: 4th International Conference on Emerging e-learning Technologies and Applications (ICETA 2005), Košice, Slovak Republic, 13 - 14 September 2005, elfa, s.r.o., 2005, pp. 165-171, ISBN 80-8086-016-6.

[7] Jakab, F.: Methods of Management Optimization and Evaluation in Computer Networks: Measurement and Evaluation of Operational Parameters, PhD thesis, Technical University of Košice, Košice, Slovak Republic, 2005, 133 pp, (in Slovak).

[8] Jakab, F., Jakab, R., Koščo, Ľ. and Giertl, J.: Communication Protocol in Computer Network Performance Parameters Measurement, In: 4th International Information and Telecommunication Technologies Symposium (I2TS 2005), Florianopolis, SC – Brazil, 2005.

[9] Jakab, F., Jakab, R., Kaščák, M. and Giertl, J.: Improving efficiency and manageability in IPFIX network monitoring platform, In: International Network Conference 2006 (INC2006), Plymouth, United Kingdom, 2006 (in printing).

[10] Jiang, W. and Schulzrinne, H.: QoS measurement of Internet real-time multimedia services. In Proceedings NOSSDAV, Chapel Hill, NC, June 2000.

[11] Klimo, M.: Voice over IP: Packet Loss and Jitter in E-model, QoS Summit'99, Paris, 1999.

[12] Quittek, J., Zseby, T., Claise, B., Zander, S., Carle, G. and Norseth, K.C.: Requirements for IP Flow Information Export, October 2004, RFC3917.

[13] Vaščák, J.: Automatic adaptation of fuzzy controllers, In: Journal of Cybernetics and Informatics, vol. 4, 2004, pp. 31-41.

[14] Hubený, T., Vodrážka, J.: Shielding Effectiveness Measuring. In RTT 2005 Proceedings [CD-ROM]. Ostrava: VŠB - TUO, FEI, Katedra elektroniky a telekomunikační techniky, 2005, ISBN 80-248-0897-8.

[15] Zseby, T., Molina, M., Duffield, N., Niccolini, S. and Raspall, F.: Sampling and Filtering Techniques for IP Packet Selection, Internet Draft, PSAMP Working Group, July 2005.

BIOGRAPHY

Juraj Giertl was born on 9.5.1980, now living in Košice. In 2003 he graduated (MSc.) at the Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, area of Computer science and informatics with focus on computer systems and networks. Now he is PhD. student at the same department. His thesis deals with optimization of measurement and evaluation of operational parameters in computer networks.

František Jakab was born in 1959. He received the MSc.degree in Systemotechnic engineering from the St. Petersburg Electrotechnical Institute (Rusia) in 1984 and the PhD. degree in 2005. He is employed as an assistant professor at the Dept. of Computers and Informatics, Technical university of Kosice. He is a head of the Computer Engineering Group and Computer Networks Laboratory (www.cnl.tuke.sk). His research interests include projecting of computer network, modeling, simulation and network management, new form of multimedia-based communication, QoS, tele-learning systems, intelligent tutoring systems and virtual universities. He has been a coordinator of several large international e-learning oriented projects supported by EC. He is a coordinator of the Cisco Networking Academy Program for the Slovak Republic and head of the Application Section of the Communication Technology Forum Association in Slovak Republic (www.ctf.sk).

Ján Bača was born in 1947. He received the Ing. (MSc.) degree in Electrical engineering from the Faculty of Electrical Engineering, Slovak University of Technology in 1968 and the CSc. (PhD.) degree in 1982. He is employed as an Associate Professor at the Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice. His current teaching and research interests are oriented towards design and diagnostics of logical circuits, CAD systems, formal specifications and decomposition of systems.

Rudolf Andoga was born on 22.8.1980. In 2003 he graduated (MSc.) at the Department of Cybernetics and Artificial Intelligence of the Faculty of Electrical Engineering and Informatics at Technical University of Ko ice. Now he is PhD. student at the same department. His thesis deals with situational control of turbojet engines (and large scale systems in general) by means of hybrid algorithms based on approaches of artificial intelligence.

Michal Mirilovič was born on 6.10.1980. In 2004 he graduated (MSc.) at the Department of Electronics and Multimedia Communications of the Faculty of Electrical Engineering and Informatics at Technical University of Košice. Now he's PhD. student at the same department. His thesis deals with continuous speech recognition in Slovak and stochastic language modeling.