International Journal of Computer Science and Software Engineering (IJCSSE), Volume 4, Issue 1, January 2015
ISSN (Online): 2409-4285 www.IJCSSE.org Page: 10-21

SOA Quality Management Subsystem on the Basis of Load Balancing Method Using Fuzzy Sets

Mykola Beshley[1], Mykhailo Klymash[2], Bohdan Strykhalyuk[3], Olga Shpur[4], Bugil Bohdan[5], Igor Kagalo[6]

[1, 2, 3, 4, 5, 6] Lviv Polytechnic National University, Lviv, Ukraine

[1]beshlebmi@gmail.com, [2]mklimash@polynet.lviv.ua, [3]bogdan_str@ukr.net

ABSTRACT

SOA quality management subsystem on the basis of load balancing method using fuzzy sets with real traffic flows on an experimental testbed has been proposed. The model of the network that is under investigation has been built. This model allows to verify functioning of prognostic traffic engineering system with evaluation of QoS parameters when deploying MPQ. In this paper the model of cloud services that is based on adaptation of the logical structure in cloud systems during migration of virtual machines has been proposed. Here, the stability of the structure in the service plane taking into account physical hardware and server resources has been assessed and overall delay of the service in the cloud system has been decreased. The modified method to numerically measure the image QoE of IPTV service by using the QoS parameters measured in the network layer and the QoS/QoE correlation analysis results on the basis of load balancing method using fuzzy sets has been created.

Keywords: *Cloud Computing, Multiservice Traffic, Fuzzy Logic, QoS, Admission Control, Resource Management, Converged Networks.*

1. INTRODUCTION

Resource allocation and quality of service management in systems based on service oriented architecture (SOA) paradigm are very important tasks, which allow to maximize satisfaction of clients and profits of service provider. In nowadays SOA systems, which utilize Internet as the communication bus the problem of service response time guaranties arises. Since overall service response time consists of communication and computational delays the task of delivering requested service response time requires proper management of both communication and computational resources. Providing effective end-to-end resource and QoS management in such complex heterogeneous converged network scenarios requires unified, adaptive and scalable solutions to integrate and co-ordinate diverse QoS mechanisms of different access technologies with IP-based QoS. Achievement of client’s requirements with the lowest costs is the essence of the problem of creation and ensuring of networks functioning. Commonly, this complex problem is divided into set of problem of smaller granularity that finally can be even more complex. One of them is the problem of resources management and traffic engineering in networks devices. Elastic solutions that are ground on evaluation and prognostication of resources state and load volumes based on correct balancing and efficient resource distribution are necessary. In order to make selection of right solution systematic tools and complex of techniques for solution of tasks of IT infrastructure support are necessary. Their creation constitutes important scientific and practical problem which solution requires deep understanding of processes that take place in convergent networks of hosting companies, i.e. IT infrastructure functioning, exact definition of specific investigation tasks, development of mathematical models and appropriate methods of tasks solving, finally realization of mentioned above tools and techniques.

We proposed a fuzzy-based resource management approach and study the performance of our implementation with real traffic flows on an experimental tested. In order to solve the problem of the existing researches, we develops the QoS/QoE correlation model to numerically evaluate the IPTV QoE by using preexisting researches’ results [1][2]. Through our proposed model, service provider can predict subscriber’s QoE in provided network environment and analogize service environment which meet the optimum QoE, conversely.

2. Fuzzy Controlled QoS System for Scalable Cloud Computing Services

Service providers offer services for transferring essential data in the form of voice, video, and data packets through a network. This information, when traversing through the network path requires different conditions of bandwidth, jitter, latency, and packet loss depending on the data type being transferred. Here arises a basic requirement for the proper allocation of resources to manage the different types of data flow. QoS, a set of mechanism, provides effective solution for managing the data flow with desired performance levels. The telecommunication operators introduce new services leading to emerge of multiservice network. Service quality assurance with services differentiation is an important task for multiservice network. It requires a concerted solution of different network resources management problems. It is necessary to implement novel traffic control algorithms for simultaneous assurance of different service quality demands with the network resources effective usage. The balancing of network node resources is related with memory buffering algorithms and services priorities association.

In this work we investigate the use of fuzzy logic for network management systems. The article suggests that network management can get huge benefit from the use of fuzzy logic in various possible applications areas. They also present a fuzzy-based method for configured resource allocation decisions. Different from, our work investigates the application of fuzzy logic to a different problem in network management i.e., dynamic resource allocation for enhanced admission control especially for tiered services within practical SOA systems. Admission control is a crucial element of the management infrastructure as it provides the means to maintain QoS of the flows already present within the network [7]. Several admission control schemes that have been previously studied are mainly classified as either parameter-based or measurement-based. While parameter-based approaches may be easier to implement as they are based on estimates of parameters such as peak rate or effective bandwidth usage in the admission request of a new flow rather than actual network measurements, measurement-based approaches provide better network utilization in the presence of bursty traffic patterns.

Proposed approach in comparison with already existing [3], uses theory of fuzzy sets. This theory is a tool for solving tasks of aggregation double-meaning, subjective and unclear evaluative judgments about state of particular parameter or indicator of optimal balancing network resources. Using classical theory of sets it is hard to connect all evaluation into one result or even sometimes in impossible. Moreover fuzzy sets can divide obtained data using linguistic terms for further decision making. Thus, if such approach is used to build a model, the latter becomes quantitative unlike existing subjective evaluations.

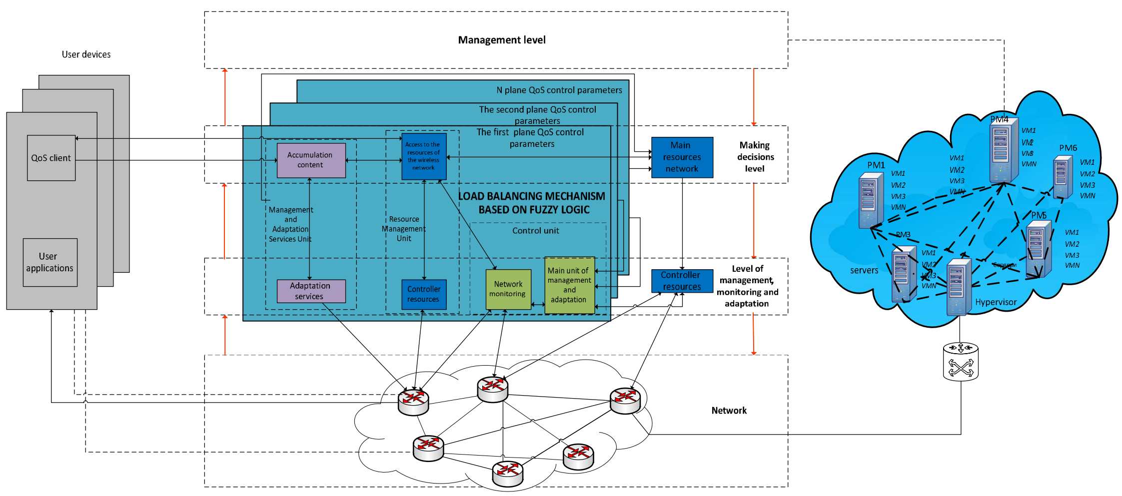

Fig. 1. Quality management subsystem on the basis of load balancing and monitoring method using fuzzy sets

We propose a SOA quality management subsystem on the basis of load balancing method using fuzzy sets with real traffic flows on an experimental testbed. Subsystem consists of distributed Policy Decision Points (PDPs) which are responsible for various management and control decisions driven by high-level declarative policies and enforced at policy enforcement points (PEPs) such as routers, switches, and gateways in the converged network transport and cloud plane.

2.1 Improving the resources efficiency distribution in the transport plane system

Commonly, for prognosticating and planning loads are used models based on assumption that data traffic corresponds to Poisson distribution. However, as results of many investigations have shown classical Poisson model is not always adequate. Leaning on a huge amount of experimental data and accurate statistical analysis it has been proved that traffic of IP networks and Internet traffic and also traffic of multiservice networks based on ATM are characterized by such correlative properties as self-similarity and long-term dependency. Thus, it is possible to prove necessity of applying self-similarity theory for building algorithms of traffic prognostication in multiservice network. Making decision concerning traffic engineering requires solution not only for task of prognostication, but also for determination of relative priorities of traffic classes. All this have to take into consideration from one side physical abilities of channels, from the other side – QoS requirements for different services. In order to ensure necessary level of quality of service for users of multiservice telecommunication network a system that is based on traffic classes determination methods [4], prognostication and load planning is proposed. Improvement of quality of service parameters for multimedia services traffic is proposed to achieve based on system of prognostic engineering that is shown on Fig.1. On the inlet of edge router there is a traffic coming from several sources. The router conducts classification of packets in order to what class does each packet belongs to. Information about intensity of arriving packets of each class is passed prognostication subsystem. The latter for traffic that belongs to self-similar traffic classes calculates prognosis on incoming load for the next time interval. Next, it calculates the size of a buffer which predicted deviation for each priority zone [5]. The problem of emerging of high risk of significant QoS decrease for low-priority flows on behalf of high-priority flows is proposed to solve by setting counter of admissible packets latency for each priority zone of buffers resource. This counter basing on analysis of four levels of latencies t1,t2, t3, t4 will mark packets by setting two reserved (not used) main bits in the DSCP field and will serve them depending on this bits combination. t1=Tadm*70% - xxxxxx00, t2= Tadm *80% - xxxxxx01, t3= Tadm *90% - xxxxxx10, t4=T Tadm *100% - xxxxxx11. Thus, packets which buffering delay reached critical level t4= Tadm*100% achieve the highest priority on the whole route to destination node and can't be buffered by the next intermediate nodes. Overview concept of prognostic engineering can be a basis of technical system that will allow decreasing packet drops probability based on prognostication. Since the channel transfers both delay-sensitive and not delay-sensitive traffic, so in order to improve quality it is proposed to dynamically set different combination in the DSCP field by changing fourth, fifth and sixth bits with different probabilities of packets drops.
In this paper the model of investigated network has been built.
This model allows verifying functioning of the system of prognostic traffic engineering. Investigated network consists of three nodes that transfer data using different algorithms of queue processing.

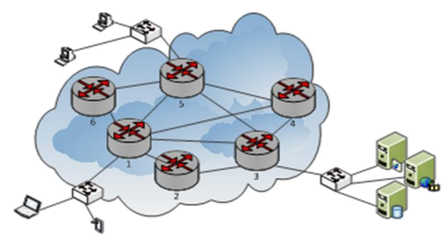

Fig. 2. The structure of the investigated network for forecasting QoS values in the transport plane system

Respectively, prognostic latency calculates by formula:
Delay – is the length of service package and is defined as the amount of time processing and batch mode service in the buffer.

$$T_{delay} = t_b + 2\frac{R_{pack}}{V} + t_{process} \tag{1}$$

where V – rate internal bus of servicing device (accepted that the rate of change of the input and output level); t_b – waiting time for packet buffer; R_{pack} – length of the packet; $T_{process}$ – duration of packet processing in the processor of servicing device; T_{delay} – duration of the service packet.

$$t_b = \sum_{j=1}^{N}\sum_{m=0}^{i-1}\left[\frac{\left[H^H (1-H)^{1-H}\sqrt{-2\ln(p)}\right]^{\frac{1}{H}} a^{\frac{1}{2H}} m^{\frac{1}{2H}}}{C_j - m}\right]^{\frac{H}{1-H}} \cdot p_{util.(i-m)j} \cdot \frac{l_{mid_{(i-m)j}} \cdot 8}{C_j} \tag{2}$$

where H – Hurst index; $P(n > N) = 10^{-6}$ – probability of packet loss, that must be ensured; m =21,05 Mbit/s – incoming packets speed (bit/s); a = 3,25 Mbit/s – coefficients of variation of incoming packets speed (bit/s).;τ_{ik} – propagation delay of i-th service priority packet in the k – channel, N – number of switching devices (nodes) located between two end subscribers of the service, C – bandwidth of the j-th channel, $p_{use.(i-m)\ j}$ – probability of using the i-th service priority in the j-th node, $l_{mid.(im)\ j}$ – the average length of i-th service priority packet in j-th node.

The probability of packet loss – defined as the number of lost packets to the total number of packets.

$$P = \frac{1}{X} \cdot \sum_{k} x_k , \tag{3}$$

where P – the probability of packet loss; X – total number of packets; x_k – number of lost packets by the k-th period.

Jitter – defined as the difference between the mean value of the delay and concrete delay.

$$J = \frac{1}{N}\sum_{i}\left|T_{average} - T_i\right|, \qquad (4)$$

where T_i - delay of i-th packet; $T_{average}$ - average value of delay.

As it can be seen from Figs. 5-7 when using proposed queues processing algorithm MPQ, that works based on prognostic network resources management using techniques of prioritization and dynamic control over delay and packets losses by changing DSCP field combinations ensures the best quality of service [5].

As it can be seen from Figs. 3-5 when using proposed queues processing algorithm MPQ, that works based on prognostic network resources management using techniques of prioritization and dynamic control over delay and packets losses by changing DSCP field combinations ensures the best quality of service.

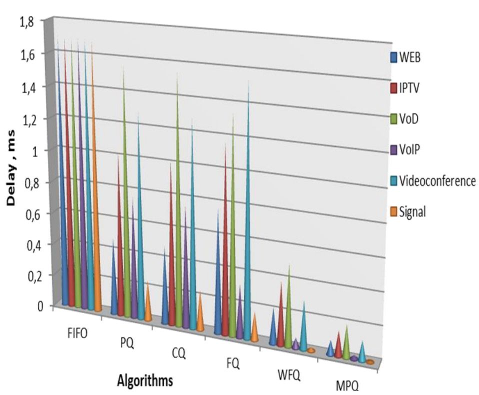

Fig. 3. Comparative evaluation of packets latencies when using different queue processing algorithms

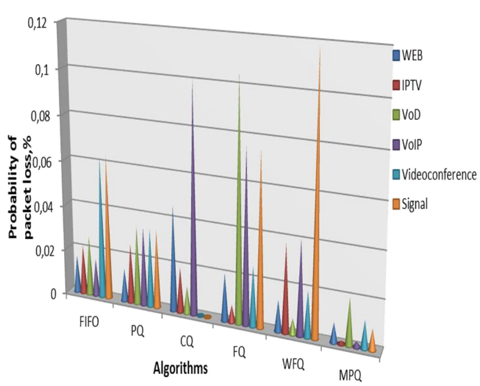

Fig. 4. Comparative evaluation of packets losses when using different queue processing algorithms

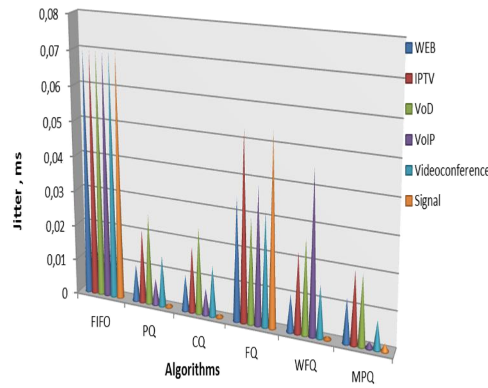

Fig. 5. Comparative evaluation of jitter when using different queue processing algorithms

2.2 Improving the resources efficiency distribution in the cloud control plane system

This paper proposes a novel network architecture called cloud control plane (Cloud C-plane). In Cloud C-plane, the forwarding functions and control functions are decoupled, and the control functions are placed in the cloud. In addition, physical network topology is modified to improve the resources efficiency when the load of traffic changes. We implement the prototype of the routing engine on an actual server, and the experimental results show the execution time of the prototype is 19 times faster in the shortest path calculation.

The advantage of cloud systems with performance distributed computing is to use of virtual machines on servers based on client requests that come to data centre and migrating VMs (Virtual Machine - VM) from one server to another.

Construction of the virtualized structure models is a pretty complicated task. The question of synthesis of logical connections arises, especially when the process of migration is traced. Given that the theories of random graphs and complex networks suit very well. In order to solve this task lets use the method of logical connections synthesis by searching central vertices based on mediation coefficient and average length of the shortest paths. The analysis of unique structures is conducted and it is investigated the question whether the vertices that are characterized by the highest degree will also have minimal average length of the shortest path L_{aver} and maximal mediation coefficient - B.

The selection criteria of the critical vertices are two parameters: mediation coefficient B and the average length of the shortest paths L_{aver} for each vertex. The length of the shortest path is also called as a distance between nodes N and M and is denoted as L(N,M). The basis for forming each of the criteria is matrix of the shortest paths. Along with determination of the mediation coefficient the number of routes that go through each edge is calculated and is represented as R matrix. The proposed method analyses the criteria and depending on the conditions satisfying processes the investigated structure in a different manner. Additionally, for each vertex the coefficient of clusterization C is calculated. This coefficient doesn't directly characterize the vertex, however, gives representation of its neighbors interconnections.

The coefficient of network clusterization is represented as a average of clusterization coefficients for all nodes in the network or is a number of triangles (nets of three nodes) that cross this node regarding to the maximal number of connections for the net of three nodes that can cross this node.

$$C_N = \frac{E_N}{(k_N \cdot (k_N - 1))} \quad (5)$$

where k_N is a number of neighbors N and E_N is a number of attached pairs between all neighbors N.

The analysis of the structures shows that selection of the vertices as a "center" of "super node" always responds to the next criteria that must simultaneously be fulfilled for each vertex: the vertex must be a central $(min(L_{aver}))$, maximally connected with other vertices $(max(k))$, and to be as a mediator between other vertices. Under these conditions supped node covers all structure with its connections. In individual cases it removal is critical and can separate from the structure individual vertex or even entire substructure. Such construction principle of virtualized structures allows tracing the migration process that is very often cased be central nodes overload.

Changing of DC structure introduces latencies to request servicing that undoubtedly will affect on the total service provisioning latency. However, considering DC structure is not enough. The important factor here is vitality of such structure, because the more this structure is stable the quicker the system processes and redirects necessary service to the user.

Delay at the interfaces appears as a result of topology changing of the network where the processing and service provisioning is held that is DC. The delay can be reduced only in case when the structure of DC is stable during the time interval T, that is the logical adaptation of request flows to the structure changes is conducted.

$$d = \frac{1}{P_{st}} \sum_{Deg.n} \frac{w_T(n)}{C - w_T(n)} \le d_{acc}, \quad (6)$$

Such change will occur not only in the service provisioning plane but also in service providers plane. All the processes in the service provisioning plane are managed by the hypervisor fig.1. Each provider must provide users with services quickly and with high quality. However, it is not worth forgetting about readiness of provider network nodes and of the physical machines of DC to process requests for service provisioning. Performance of the node will be characterized by the presence of necessary parameters and CPUs that will process the request. Parameters of server performance where the M_i virtyual machines (VM) are located can be calculated by formulas[6]:

$$CPU_{pr} = \frac{\sum_{i=1}^{k} M_i \times CPU_i}{\sum_{i=1}^{k} M_i}, \quad (7)$$

$$RAM_{pr} = \frac{\sum_{i=1}^{k} M_i \times RAM_i}{\sum_{i=1}^{k} M_i}, \quad (8)$$

де M_i – quantity VM i - th PM (physical machine); CPU_i – clock rate VM i-th PM; RAM_i – – random access memory of VM i-th PM у; k – quantity PM cloud system.

The stability of cloud system structure based offer to a method that includes an assessment of the network structure (network model), load each virtual machine at a time availability T, readiness coefficients of node and nodes interconnectedness with each other [7]

$$P_{st} = 1 - \sum_{i=1}^{n} w_i(t) + K_r + P_c, \quad (9)$$

де $w_i(t)$ – load of nodes at time t; P_c – probability of connectivity nodes; K_r – readiness coefficients of node

The total delay of the service to the end user subsystem is defined as the sum of the time of delay in the transport plane and the time delay in cloud plane when the structure of data center reconfiguration:

$$D = T_{delay} + d \quad (10)$$

3. Method of Load Balancing Using Fuzzy Logic

Load balancing of resources is critical to the success of a cloud environment, as well as other environments such as traffic over a transport network. Load balancing may take into account different considerations such as, for example, server processing speed of a single server or over an entire grid, available memory, I/O bound processes, the tasks that are running on the grid, etc. Usually the aim of load-balancing is to move the running tasks across the server processing units in order to insure that no processor is idle or overworked during run time. Said otherwise, load balancing seeks to optimally balance the load over an entire grid in cloud plane subsystem. In theory, load balancing should minimize the total running time by a set of tasks. The propose method generally relates to a system and mechanism of load balancing using fuzzy logic and, more particularly, to system and method of load balancing tasks over a cloud environment including, for example, optimizing server utilization, traffic over a network and other functions. The decision can be implemented over any distributed network or stand-alone server, for example. By using the method, it is possible to efficiency and continuously balance loads over a SOA environment.

The idea of the method is following. In heterogeneous networks based on SOA, degree of node load level due to load balancing is equal as in transport network so and by the level of utilization of transport resources. However, criteria can include each other opr eve worth be contradictive one to another. This way, there is a necessaty of special methods of requests distribution by particular multicriteria analysis of the state of all nodes in the subsystem. Fuzzy logic can be used to make this analysis easier. It is possible through forming of the number of criteria where each has its weight. For each criteria, good, bad and intermediate values are defined. Then the values of the criteria are introduced using triangle fuzzy numbers. This number are multiplied by the coefficients in order to find general value (both for nodes adn the whole network). The next step is, using defuzzyfication the point is determined on the scale that characterizes respective element of the system or the whole system and gives adequate evaluation. Then due to fuzzy logic at the border of two states we can make a decision which state is closer despite the contraditions in the system (e.g., good or bad, basedon the value of respective function in respective point). After obtaining evaluations, we can make adequate judgements about increasing load intensity on some server. The weights is proposed to form dynamically with some intervals in recalculation by type:

In case of service platform node utilization level increase it is decreasing of the weigth of criteria "request processing time - system response time", "percent of refused requests", "time of requests to DB", "percepted quality of service".

In case of service platform node utilization level decrease it is increasing of weigth of criteria "request processing time - system response time", "percent of refused requests", "delay".

After preparing parameters for method of load balancing using fuzzy logic in subsystem SOA, a scale of evaluation of importance of each criteria for load balancing is introduced in accordance with Likert scale (1 – very inapplicable (0,0,0.25), 2- inapplicable (0,0.25,0.75), 3 – medium, (0.25, 0.5, 0.75), 4 – applicable (0.5,0.75,1), 5 – very applicable (0.75,0.75,1)) according to the Table I. A five-point scale of linguistic terms is also introduced for evaluation of correlation between state at the enterprise and criteria (1 – “very low”, 2 – “low”, 3 – “average”, 4 – “high”, 5 – “very high”), that is evaluation of parameter index level regarding to respective criterion in accordance with mention scale .

In the considering SOA network it is deployed seven services such as voip (A), and video conference (B), and IPTV (C), and Internet data (D), and interactive data (E), and media on demand (F), and FTP (G)).

Let’s form empiric coefficients of services importance regarding required quality of service for load balancing using coefficients B_1, B_2, B_3, B_4 that can be in a range from 1 to 3, where higher value means higher importance of specific quality of service parameter for a particular service category for transport plane SOA system and K_1, K_2, K_3 and K_4 for cloud plane SOA system (see Table III).

Table 1: Parameters evaluation for load balancing process scale in transport plane with Likert

SCALE	1	2	3	4	5	The Min/Max Quantities
Load, [%]	70–100	50 –70	40 –50	30–40	0–30	0–100
Delay,[ms]	<300	200–300	100–200	50–100	10–50	10–500
Jitter, [ms]	<30	20–30	10–20	5–10	1–5	10–30
Packet Loss Ratio (PLR), [%]	<8	6–8	4–6	3-4	1-3	1-8
Throughput, [Mbps]	<50	50-100	100-200	200-500	500-800	800-1000

Table 2: Parameters evaluation for load balancing process scale in cloud plane with Likert

SCALE	1	2	3	4	5	The Min/Max Quantities
Download CPU	<85	85-70	70-60	60-50	50-40	40-100%
Download RAM	<80	80-70	70-60	60-50	50-40	40-100%
Time delay of service[mks]	<500	450-400	400-300	300-200	200-100	500-100
Loss probability of request, [%]	<1	0.9-0.8	0.8-0.7	0.7-0.6	0.6-0.5	0.5-1%
Server load	<100	80-60	60-40	40-25	25-10	10-100%

Table 3: Services and respective traffic parameters significance coefficient relative to others

P \ S	PLR B_1	Delay B_2	Jitter B_3	Throughput B_4	CPU K_1	RAM K_2	Time delay of service request K_3	Loss probability of request K_4
A	2	3	3	1	1	1	2	3
B	2	3	3	2	2	2	3	2
C	3	2	2	3	3	2	2	1
D	3	1	1	1	1	1	3	1
E	2	2	1	1	2	2	3	2
F	2	2	2	3	2	2	1	3
G	3	2	1	1	2	2	2	3

After values of node parameters were estimated in the process of node evaluation based on collection of indexes that characterize QoS by default with all weights equal to 0.5. In case of weight decreasing while ρ_{mn} increasing, weight of parameter is calculated in accordance to Table IV by formula:

In case of weight increasing while ρ_{mn} increasing, weight of parameter is calculated in accordance to Table I-II by formula:

$$wp_{imn} = 0.5 + 0.5 \cdot \rho_{mn_{transport}}\ ,\ i = 1..4 \tag{11}$$

$$plp_{kmn} = 0.5 + 0.5 \cdot \rho_{mn\ cloud}\ ,k = 1..4 \tag{12}$$

where i is an index number of QoS-affecting parameter transport plane wp_{imn} by the order (see Table II) that is taken into account in the process of load balancing and k is an index number of cloud plane parameter plp_{kmn} that is taken into consideration in the process of of load balancing (see Table III).

Finally, weights w_{imn} are obtained as a result of multiplying parameters' weights (1-2) and coefficient of parameter importance regarding to requested service:

$$w_{imn} = wp_{imn} B_i\ , i = 1..4, \tag{13}$$

$$pl_{kmn} = plp_{kmn} K_k\ , k = 1..4, \tag{14}$$

where B_i is coefficient of transport plane parameters importance regarding to service type requested by user

(shown in Table II) and K_k is an importance coefficient of cloud plane parameters subsystem SOA (shown in Table III).

Thus normalization of w_{imn} is performed:

$$W_{imn} = \frac{w_{imn}}{\sum_{i=1}^{4} w_{imn}}\ , \tag{15}$$

Table 4: Services and respective traffic parameters significance coefficient relative to others

	Load changes	Weight changes
Probability of disconnection [%]	Increasing ρ ↑	Increasing ↑
Delay,[ms]	Increasing ρ ↑	Increasing ↑
Jitter, [ms]	Increasing ρ ↑	Increasing ↑
Packet Loss Ratio (PLR), [%]	Increasing ρ ↑	Increasing ↑
Throughput, [Mbps]	Increasing ρ ↑	Increasing ↑
Download CPU	Increasing ρ ↑	Increasing ↑
Download RAM	Increasing ρ ↑	Increasing ↑
Time delay of service reguest[mks]	Increasing ρ ↑	Increasing ↑
Loss probability of request, [%]	Increasing ρ ↑	Increasing ↑

Respectively, TFN evaluation of selected access node $\widetilde{Q}_{mn}$ is calculated regarding to selected the best path streaming in accordance with parameters evaluation scaling listed in Table I, that immediately affect QoS for respective user serving:

$$\widetilde{Q}_{mn} = (q_1, q_2, q_3)_{mn} = \sum_{i=1}^{4} \left(W_{imn} \times \widetilde{L}_{imn}\right) \quad (16)$$

$$q_{j_{mn}} = \sum_{i=1}^{4} \left(W_{i_{mn}} \times l_{ij_{mn}}\right), \quad (j = 1,2.3;\ m = 1,2...,6;\ n = 1), \quad (17)$$

where q_1, q_2, q_3 are bottom level of general evaluation $\widetilde{Q}$, its basic value and top level respectively, $\widetilde{L}_{imn} = (l_{i1}, l_{i2}, l_{i3})_{mn}$ - triangular fuzzy number that characterizes indicator of node parameter by *j-th* criteria for node of transport plane system - *m* based on *n-th* technology. Here l_{i1}, l_{i2}, l_{i3} - bottom level of linguistic variable, its basic value and top level respectively to TFN format (Triangular Fuzzy Number [3]), see scale Fig. 5.

Then normalization of w_{kmn} is performed:

$$PL_{kmn} = pl_{k\,mn} \Big/ \sum_{i=1}^{4} pl_{k\,mn}, \quad (18)$$

Respectively the selected node evaluation $\widetilde{P}_{mn}$ is calculated regarding to selected the best server for process services according to parameters evaluation that are obtained from Table II in cloud plane:

$$\widetilde{P}_{mn} = (p_1, p_2, p_3)_{mn} = \sum_{i=1}^{4} \left(PL_{kmn} \times \widetilde{L}_{kmn}\right) \quad (19)$$

$$p_{j_{mn}} = \sum_{i=1}^{4} \left(PL_{k_{mn}} \times l_{kj_{mn}}\right), \quad (j = 1,2,3;\ m = 1,2,...6;\ n = 1), \quad (20)$$

where p_1, p_2, p_3 are bottom level of general evaluation $\widetilde{P}$, its basic value and top level respectively, $\widetilde{L}_{kmn} = (l_{k1}, l_{k2}, l_{k3})_{mn}$ is triangular fuzzy number TFN that characterizes indicator of node parameter by *k-th* criteria for node of *m* cloud plane system based on *n-th* cluster. Here l_{k1}, l_{k2}, l_{k3} is bottom level of linguistic variable; its basic value and top level respectively to TFN format (see Fig.5).

Let's conduct defuzzification of obtained fuzzy number for transport plane system (13) in accordance with [3] and cloud plane [d]:

$$Q_{mn} = 1/3 \times \sum_{j=1}^{3} q_{jmn}. \quad (21)$$

$$P_{mn} = 1/3 \times \sum_{j=1}^{3} p_{jmn}. \quad (22)$$

4. Investigation SOA Subsystem on the Basis of Load Balancing Imitation Model Using Fuzzy Logic

Imitation modeling of serving process always required for developer of imitation model as well as testing of adequacy of created model to processes that are performed in a real system. The simplest way of determining characteristics of service system lies in obtaining of experimental data regarding to serving process. Data analysis allows determining what parameters of service system must be changed in order to increase quality of service, i.e. optimize the process.

Existing service systems contain wide variety of components where each component is a complex system that has its parameters and characteristics. In general all components individually affect quality of service of the whole system. That is why for creating adequate model and performing adequate evaluation of modeling results it is necessary to take into account all components that take part in service process.

Big number of users, applications and sessions that are generated by these applications and their variety have significant impact on characteristics of traffic that arrives at service system. Thus, in order to model such traffic it is necessary to apply mathematical apparatus that will allow describing characteristics of such traffic more precisely. In is understood that the most effective way of modeling in such situation is development of specialized software.

By virtue to software realization of imitation model it is possible to realize not only all necessary functions of model but also ensure control over its work. Software allows using graphical user interface dynamically change model parameters and by that evaluate systems behavior in specific situation that can occur in a real service system. Besides that, software using GUI allows in a real time all parameters of the model, what can be done using graphics, diagrams, lists and tables.

For investigation of mobile network functioning with high mobility if users it is necessary to develop imitation model using big number of parameters and characteristics that allow describing existing networks using mathematical, prognostic and optimization models. Thus, in this paper a model SOA quality management subsystem on the basis of load balancing method using fuzzy sets is developed. For developing such software in this paper is used programming language C++ based on development environment Borland Builder C++ 6.0.

4.1 Experimental Results

The most loaded node is marked with red color. Subsystem with fuzzy logic sets for it the worst TFN index, as a result of load balancing method the migration of request flows for service provisioning (the entire virtual machine) takes place and the logical structure of cloud network changes. Due to newtwork structure change with respect to method of its construction the subsystem recalculates all critical moments of such system and sets respective indexes of quality TFN to each physical node. As a result of applying of such principle the system can redefine the most loaded server and unload it without significant negative impact on service provisioning. In the transport plane, after the fuzzy indexes of quality TFN for routers have been determined, routing is conducted based on the proposed method of load balancing and will point to the optimal transmission paths.

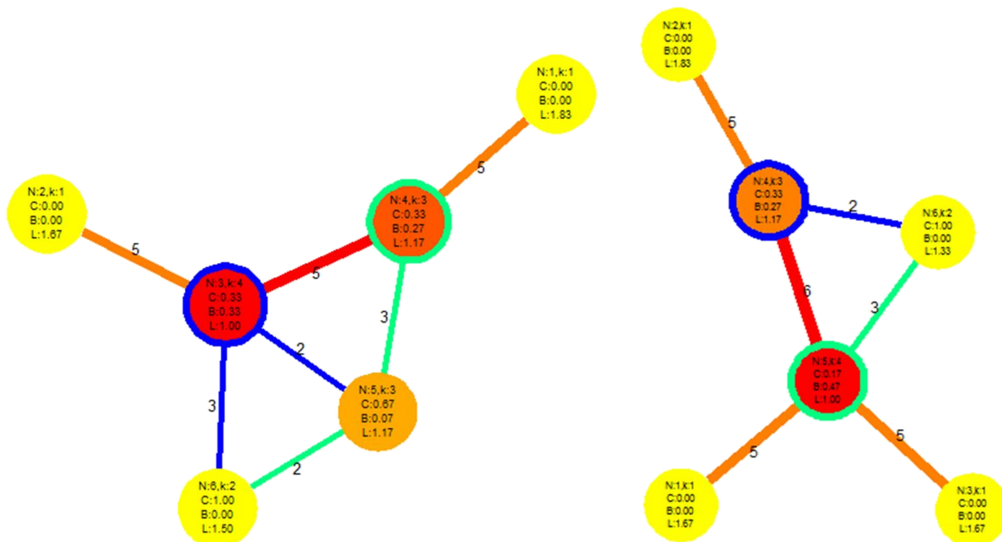

Fig. 6. Unloading the node and performing service migration after the load balancing method has been applied

We propose a SOA quality management subsystem on the basis of load balancing method using fuzzy sets with real traffic flows on an experimental testbed. Subsystem consists of distributed Policy

Table 5: Fuzzy numerical evaluation for nodes of transport plane subsystem and its defuzzification

	Router1 ($\tilde{Q}_{mn}$)	**Router2** ($\tilde{Q}_{mn}$)	**Router3** ($\tilde{Q}_{mn}$)	**Router4** ($\tilde{Q}_{mn}$)	**Router5** ($\tilde{Q}_{mn}$)	**Router6** ($\tilde{Q}_{mn}$)
Probability of disconnection	(0,2; 0,005; 0,005)	(0,2; 0,005; 0,005)	(0,105;0,005; 0,005)	(0,2; 0,005; 0,005	(0,15; 0,005; 0,005)	(0,105; 0,005; 0,005)
Delay	(0,04; 0,053; 0,03)	(0,04; 0,053; 0,03)	(0,008; 0,053; 0,003)	(0,04; 0,053; 0,03)	(0,04; 0,053; 0,023)	(0,04; 0,053; 0,008)
Jitter	(0,15; 0,095; 0,075)	(0,1; 0,0095; 0,06)	(0,01075;0,0057; 0,1750)	(0,25; 0,195; 0,175)	(0,25; 0,0095; 0,175)	(0,01; 0,005; 0,175)
Packet Loss Ratio	(0,105; 0,04)	(0,05; 0,105; 0,04)	(0,1; 0,105; 0,0055)	(0,1; 0,105; 0,04	(0,1; 0,105; 0,0055)	(0,1; 0,105; 0,0055)
Throughput	(0,0025; 0,0125; 0,01)	(0,0025; 0,0125; 0,01)	(0,006; 0,011; 0,001)	(0,0051; 0,0225; 0,01)	(0,006; 0,0225; 0,001)	(0,006; 0,006; 0,001)

Load	(0,00525;0,00525; 0,00525)	(0,00525;0,00525; 0,00525)	(0,00525;0,00525;0,00525)	(0,00525;0,00525; 0,00525)	(0,00525;0,00525; 0,00525)	(0,00525;0,00525; 0,00525)
$\widetilde{Q}_{mn}$	(0,51475;0,43275; 0,56025)	(0,51475;0,29725; 0,29275)	(0,36895;0,2555; 0,516)	(0,78985;0,69275; 0,86025)	(0,71825;0,52475; 0,70975)	(0,46075;0,48125; 0,69475)
Q_{mn}	**0,402583333**	**0,26825**	**0,28015**	**0,68095**	**0,5509166**	**0,445583333**

Table 6: Fuzzy numerical evaluation for nodes of cloud plane subsystem and its defuzzification

	Server PM1 ($\widetilde{P}_{mn}$)	Server PM2 ($\widetilde{P}_{mn}$)	Server PM3 ($\widetilde{P}_{mn}$)	Server PM4 ($\widetilde{P}_{mn}$)	Server PM5 ($\widetilde{P}_{mn}$)	Server PM6 ($\widetilde{P}_{mn}$)
Download CPU	(0,0025; 0,0125; 0,01)	(0,0025; 0,0125; 0,01)	(0,006; 0,011; 0,001)	(0,0051; 0,0225; 0,01)	(0,006; 0,0225; 0,001)	(0,006; 0,006; 0,001)
Download RAM	(0,0225; 0,0525; 0,15)	(0,0225; 0,0525; 0,07)	(0,00375;0,05375; 0,15375)	(0,0325; 0,0525; 0,25)	(0,0325; 0,075; 0,15)	(0,0375; 0,0525; 0,15)
Time delay of service reguest	(0,032; 0,017; 0,12	(0,032; 0,017; 0,02)	(0,0052; 0,012; 0,12)	(0,032; 0,017; 0,12)	(0,032; 0,012; 0,12)	(0,032; 0,012; 0,12)
Loss probability of request,	(0,0125; 0,0875; 0,125)	(0,0625; 0,0375;0,052)	(0,125;0,00475; 0,0475)	(0,125; 0,2375; 0,225)	(0,1025; 0,2375; 0,225)	(0,125; 0,2375; 0,225)
Server Load	(0,00525;0,00525; 0,00525)	(0,00525;0,00525; 0,00525)	(0,00525;0,00525;0,00525)	(0,00525;0,00525; 0,00525)	(0,00525;0,00525; 0,00525)	(0,00525;0,00525; 0,00525)
$\widetilde{P}_{mn}$	(0,51475;0,43275; 0,56025)	(0,51475;0,29725; 0,29275)	(0,36895;0,2555; 0,516)	(0,78985;0,69275; 0,86025)	(0,71825;0,52475; 0,70975)	(0,46075;0,48125; 0,69475)
P_{mn}	**0,30583333**	**0,36825**	**0,2015**	**0,48095**	**0,509166**	**0,545583333**

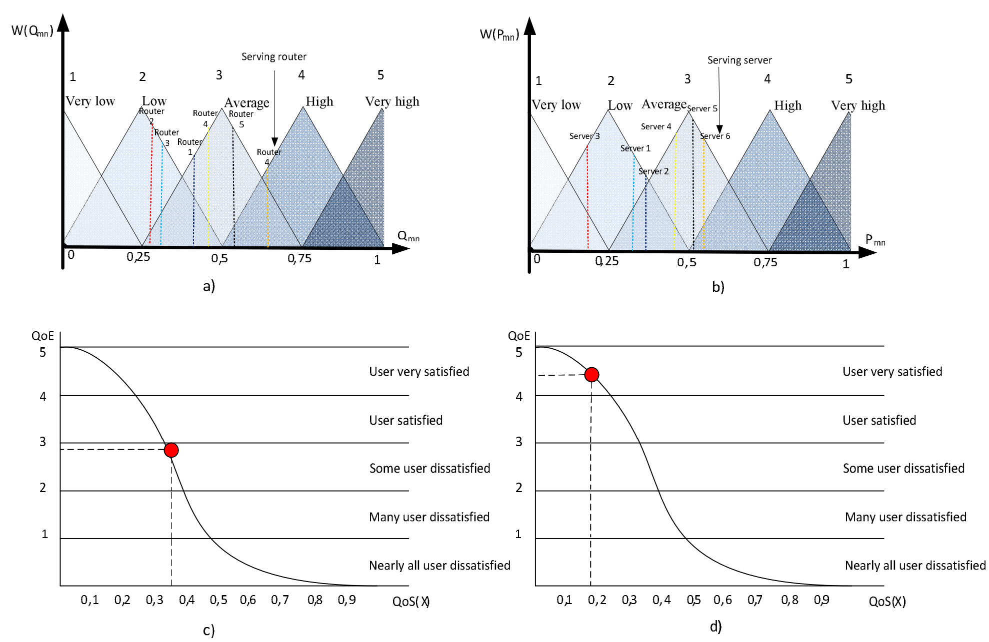

Fig.7 Optimal router(a) and server PM (b) determination by using Likert scale and image QoE measurement example of the IPTV service (c) before use the method of load balancing and (d) after application of the method of balancing.

5. Conclusions

In this paper has been proposed a fuzzy-based resource management approach and study the performance of our implementation with real traffic flows on an experimental tested. Subsystem consists of distributed Policy Decision Points (PDPs) which are responsible for various management and control decisions driven by high-level declarative policies and enforced at policy enforcement points (PEPs) such as routers, switches, and gateways in the converged network transport and cloud plane. Proposed approach in comparison with already existing, uses theory of fuzzy sets. This theory is a tool for solving tasks of aggregation double-meaning, subjective and unclear evaluative judgments about state of particular parameter or indicator of optimal balancing network resourses. We propose the modified method to numerically measure the image QoE of IPTV service by using the QoS parameters measured in the network layer and the QoS/QoE correlation analysis results on the basis of load balancing method using fuzzy sets. Through our proposed model, network providers can predict subscriber's QoE in provided network environment and analogize service environment which meet the optimum QoE on the contrary. On a real time basis, it is more rapidly able to correspond to the poor quality by monitoring the QoE of the IPTV service. The service provider can provide the multimedia service of the improved QoE through the proposed QoE control processes. And moreover, the network operator can prevent the unnecessary investment for the enlargement, maintenance and repair of the network.

REFERENCES

[1] Kwang-Jae Kim, Wan-Seon Shin, Dae-Kee Min, Hyun-Jin Kim, JinSung Yoo, Hyun-Min Lim, Soo-Ha Lee, and Yong-Kee Jeong, "Analysis of key features in IPTV service quality model," IEEM2008, vol.1, pp.595-598, Dec. 2008.

[2] Kwang-Jae Kim, Wan-Seon Shin, Dae-Kee Min, Hyun-Jin Kim, JinSung Yoo, Hyun-Min Lim, Sook-Ran Lee, and Yong-Kee Jeong, "Service Quality Model for IPTV Service: Identification of Key Features and Their Relationship," INFORMS2008-Service Science, Washington D.C, Oct. 2008.

[3] Klymash M., Stryhaluk B., Demydov I., Beshley M., Seliuchenko M "A Novel Approach of Optimum Multi-criteria Vertical Handoff Algorithm for Heterogeneous Wireless Networks" International Journal of Engineering and Innovative Technology (IJEIT)Volume 5, Issue 5, November 2014 p. 41-52.

[4] M.Beshley, T. Maksymyuk, B. Stryhaluk, M. Klymash. Research and Development the Methods of Quality of Service Provision in Mobile Cloud Systems. IEEE International Conference [Black Sea Conference on Communications and Networking (BlackSeaCom'2014)], Odessa, Ukraine, May 27-30, 2014, P. 165-169.

[5] Mykhailo Klymash, Mykola Beshley, Orest Lavriv. Model of network resources management on the basis of services priorities association. Proceedings of international conference CADSM'2013. Polyana-Svalyava. – 2013. p. 172-173.

[6] M. Klymash, M. Beshley, B. Buhyl, V. Romanchuk. "Method of Resource Distribution for Mobile CloudComputing Systems", Conference [Modern Problems of Radio Engineering Telecommunications and Computer Science (TCSET)], 19-23 February, Lviv-Slavsko, 2014, P. 581-584.

[7] Yerima, S.Y.; Parr, G.P.; McClean, S.; Morrow, P.J. Measurement-based Policy-driven QoS Management in Converged Networks. In Proceedings of the IEEE National Communications Conference, NCC 2011, Bangalore, India, 28–30 January 2011.

AUTHOR PROFILES:

Mykola Beshley is now PhD student at Telecommunications department, Lviv Polytechnic National University, and received his M.S. degree in information communication networks from Lviv Polytechnic National University in 2012. His research interests include converged networks, mobile cloud computing, machine-to-machine communication and heterogeneous networks, software defined radio access networks.

Mykhailo Klymash is now the Chief of Telecommunication Department, Lviv Polytechnic National University, Ukraine. He received his PhD in optical data transmission, location and processing systems from Bonch-Bruevich Saint-Petersburg State University of Telecommunications, Saint Petersburg, Russia, in 1994 Honored member of Ukrainian Communications Academy . The topics of his current interest of research include distributed networks, cloud computing, convergent mobile networks, big data, software defined networks and 5G heterogeneous networks.

Bohdan Stryhalyuk PhD, Postdoctoral Fellow, Telecommunications department, Lviv Polytechnic National University. He received his PhD in telecommunication systems and nets from Lviv Polytechnic National University in 2009. Scientific interests: theoretical foundations of telecommunications networks analysis and synthesis on the basis of Cloud technologies, efficiency increasing for multiservice traffic management, efficiency increasing for wireless communications.

Olga Shpur is now PhD student at Telecommunications department, Lviv Polytechnic National University, and received his M.S. degree in information communication networks from Lviv Polytechnic National University in 2013. Here research interests: include design features and operation of networks based on service-oriented architecture, mesh- and cloud-technology.

Bohdan Bugil is now Associate Professor at Telecommunications department, Lviv Polytechnic National University. He received his PhD in telecommunication systems and nets from Lviv Polytechnic National University in 2010. Scientific interests: concurrent data processing, big (meta) data capturing, efficiency increasing of deploying and research methods for NGN, operating modeling and research for multiservice networks, the use of math and statistic modeling for telecommunication purposes.

Igor Kagalo is now PhD student at Telecommunications department, Lviv Polytechnic National University, and received his M.S. degree in information communication networks from Lviv Polytechnic National University in 2011. His research interests include 5G wireless communication networks, cloud computing, inter-cell interference management.