Simulating E6 Networks Dynamic Routing

Guliaiev K.D., Zaitsev D.A.
Odessa National Academy of Telecommunications
Kovalska, 1, Odessa 65029, Ukraine
Email: k.guliaiev@gmail.com, zsoftua@yahoo.com

Abstract–**Principles of known dynamic routing protocols adaptation into E6 networks were presented. Specifications and components for the distance-vector protocol E6-RIP modeling were developed in the language of colored Petri nets. The simulation of E6 networks series under the devices turning off was implemented for the evaluation of the protocol robustness and its parameters choice. The protocol provides fast adaptation to the variable network structure.**

Keywords: E6 technology; dnamic routing; colored Petri net; simulation; variable network structure

I. INTRODUCTION

E6 technology [1,2] constitutes a new approach to the organization of the telecommunication networks, which are completely constructed on the base of Ethernet. Modern tendencies of the telecommunications development are aimed to the dominance of Ethernet either on periphery or on backbones. But under the conditions of homogeneous medium the duplicate addressing IP-MAC and protocols TCP, IP themselves are redundant and decrease considerably the efficiency of the network functioning.

IEEE attempts to provide the scalability of Ethernet via Provider Backbone Bridge technology (PBB) [3] lead to the frame headers enlargement and bring an additional mapping of backbone and customer MAC addresses.

The advantages of E6 technology [1,2] are the following: expansion of the network address space, annulment of protocols TCP, IP and addresses mapping ARP/RARP as well as the corresponding packet headers, speed-up of the working algorithms of the network devices.

The simulation of E6 networks by colored Petri nets using simulating system CPN Tools [5] was implemented [4] to confirm their robustness. But at the models construction on the base of modified components [6], static routing tables were used.

The goal of the present paper is the forming of adaptation principles for using known dynamic routing protocols within E6 networks as well as the construction of distance-vector routing algorithms components for E6 networks.

II. E6 NETWORKS ROUTING PECULIARITIES

E6 address constitutes a uniform 6 octet hierarchical address, which is used on all the levels of the ISO open systems interconnection model. Similar to CIDR IP address it consists of subnet address and host address with the border given by the subnet mask (subnet address bits number). The length of 6 octets provides the direct placement of E6 address into the MAC address fields of the Ethernet frame header. User interfaces are not changed under the employment of E6-DNS.

At data-link and physical layers, the possibility of the device MAC-address substitution via software is used for the assigning E6 address; for guaranteed delivery of information (similar to TCP), Ethernet LLC2 is employed. The basic network device is a switching-router SRE6.

The distinctive feature of the frame (packet) delivery process within E6 network is the immutability of the frame header, which contains the pair of E6 destination and source addresses; the key information at the route choice is E6 destination address.

The network is formed by SRE6 connected to each other as well as to terminal (customer) devices. The address table of SRE6 has the following format:

(e6nw, metric, port, options),

where e6nw=(E6A,mask) is the identifier of E6 network (subnet), E6A – E6 host/subnet address, mask – the subnet mask in the form of the subnet address bits number, metric – the metric of the route, port – the port number for the frame forwarding, options – additional parameters.

The pair (E6A,mask) can specify either E6 subnet or a separate host (under mask=48); in this case the mask gives the number of bits of E6 destination address, which have to be compared at the route choice. At the alternative routes choice, usual preferences are used: the largest mask, the least metric.

The advantages of SRE6 in the comparison with traditional IP-routers are the following: the absence of duplicate packet encapsulation into frame, the absence of IP-MAC addresses mapping, the indication of the interface and gateway only via physical port number. Listed peculiarities lead to simplification of dynamic routing algorithms. In the comparison with Ethernet switches, SRE6 provides the aggregating E6 addresses under common mask and in this way decreases considerably the number of address table records allowing the world-wide networks construction.

The way of known dynamic routing algorithms (protocols) adaptation mainly from TCP/IP stack of protocols is proposed. The choice is stipulated by the hierarchical structure of E6 address similar to IP address. The alternative of the passive listening and broadcasting accepted in Ethernet (and PBB) was refused since it decreases the effective network performance in spurts and leads to the temporary overload.

The principles of TCP/IP dynamic routing algorithms adaptation into E6 networks can be formulated in the following way:

- extend the address field from 4 to 6 octets and enlarge the legal mask length from 32 to 48 bits;
- replace the interface IP-address and gateway IP-address by the SRE6 physical port number;
- modify the message formats and algorithms of their forming, transmitting, processing.

III. PROTOCOL E6-RIP

In the present work the adaptation of the known distance-vector dynamic routing protocol RIP [7] was implemented. The results are represented in the language of colored Petri nets that allows their direct usage in the network models within the environment of CPN Tools [5] as well as in further software-hardware realization of SRE6.

The work of the protocol is based on the periodical sending of complete routing table to the neighbors (with period uta) and immediate (with period tuta) sending of trigger changes at the routes update. For the route loops avoidance, the splitting horizon method is employed when the update is not sent to the neighbor, which it was received from.

Using sophisticated metrics is possible, but in the majority of cases the number of intermediate nodes (hops) is considered as a metric. The maximal metric value INFINITY is chosen, which limits the valid route length. Networks (subnets) with the metric INFINITY (and greater) are considered not reachable.

The E6-RIP message has the following format:

(operation, e6nw, metric),

where operation is one of the valid operations: REQUEST, RESPONSE. Regular updates are sent on timer uta in the response format even if the corresponding request was not received.

Within the routing table, the following options required for the protocol functioning were allotted:

(e6nw, metric, port, chg, ta, gcta),

where chg – the indication of changes, ta – the record ageing timeout, gcta – the garbage collection timeout.

Timeout ta is reset at the route receiving from a neighbor; at the lapse of ta, the record is marked as the unreachable network by the metric INFINITY and gcta is set on. At the lapse of gcta the record is erased out of the table.

At the update receiving the metric is incremented by the cost of the corresponding channel (by unit). A new route is added into the table; a new route to the unreachable network (metric=INFINITY) is ignored. The processing of the known network update is implemented in the following way:

- replace the record at the lesser metric;
- set up the metric INFINITY and set on gcta at the receiving of INFINITY for the record with the metric lesser than INFINITY;
- ignore the receiving of INFINITY for the record with the metric INFINITY;
- ignore the update with the greater metric (or with the equal metric and other port number);
- reset ta at the receiving the update with the equal metric and the same port number.

The indicator of changes chg is used for the subsequent sending the trigger changes; it is set on at the record adding and metric change. The anticipatory saving of an alternative route at the equal metric and elapsing timeout ta is an option.

The RIP standard [7] recommends the following values of timers: uta=30 s., ta=180 s., gcta=120 s., tuta=1-5 s., which were the subject of analysis for E6-RIP. The option of the splitting horizon with poisoned reverse, when the update is forwarded to the sender with the metric INFINITY, was studied. Moreover, the value INFINITY itself was analyzed (the standard value is equal to 16).

IV. SRE6 MODEL

The model of IP-router [6] was chosen as the base, which was modified regarding E6 addresses usage, the port model was put into separate page (similar to [4]), and also recursive functions [8] were employed for the work with address table that is represented by the list structure.

The SRE6 model is assembled by cloning the required number of ports port supplied with the components RIPprocess, RIPupdate and the places modeling the internal memory objects. The descriptions of basic data types (color sets) are listed in Fig. 1. En example of the model for four ports is represented in Fig. 2.

The places in* give the numbers of corresponding ports. The places p*IN, p*OUT model the port input and output channels correspondingly. The SRE6 internal memory is represented by the following places: RT – the routing table, Buf – the frames (packets) buffer, msg-in – the E6-RIP input message buffer, Bufrt – the E6-RIP output messages buffer. The routing table RT is modeled by the contact (I/O) place for the indicating at the model main page.

```
colset e6=product INT*INT*INT*INT*INT*INT;
colset mask=INT;
colset nwt=product e6 * mask timed;
colset b=INT timed;
colset pkt=record e6src:e6 * e6dst:e6 * data:b * dt:INT;
colset operation=with REQUEST | RESPONSE;
colset rtm=record opr:operation*nw:nwt *m:INT;
colset cha=union pk:pkt+rm:rtm timed;
colset rtmin=record opr:operation*nw:nwt
*m:INT*ifn:INT;
colset rtr=record nw:nwt *m:INT*ifn:INT*chg:BOOL*
ta:INT*gcta:INT;
colset buf=product pkt*INT timed;
colset brtm=product rtm*INT;
colset brtr=product rtr*INT;
```

Figure 1. Description of basic data types (color sets)

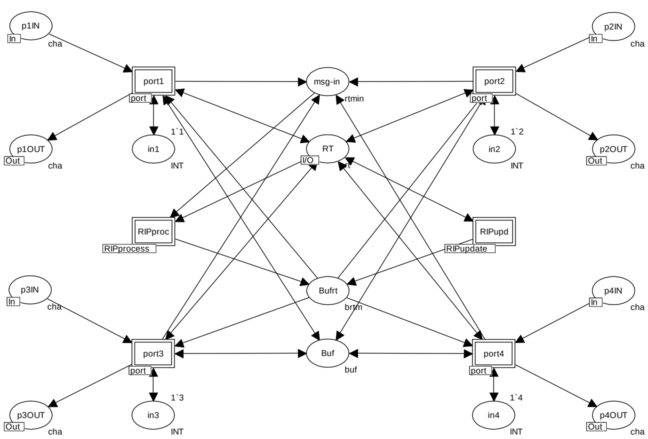


Figure 2. SRE6 model (4 ports)

The SRE6 port model (port) is represented in Fig. 3. The port regular work is provided by the pair of transitions getpkt, putpkt, which implement the input packet receiving with storing into the buffer Buf and the packet transmitting from the buffer correspondingly. At the packet storing into the buffer, the port for the packet forwarding is determined on the routing table RT. Recursive function grec finds the appropriate routing table record; the function sameNW implements the routing table E6 address comparison with the packet destination address (e6dst); the table records with the metric INFINITY are ignored. The transition droppkt models the dropping of packets at the address information absence meanwhile the counter ndrop is incremented.

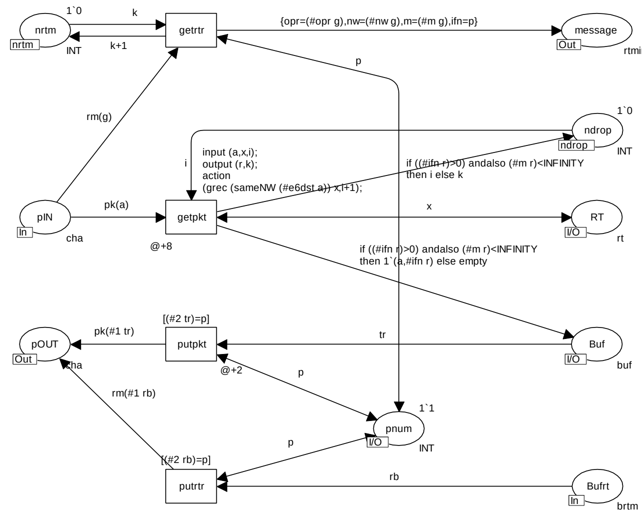


Figure 3. SRE6 port model (port)

The pair of transitions getrtr, putrtr models the E6-RIP messages receiving and sending correspondingly. The calculation of the received E6-RIP messages is implemented via the counter nrtm. For the channel information discerning the union data type is used; the indicator pk distinguishes the regular packets, the indicator rm – the E6-RIP messages.

Let us muse on the basic model data types' descriptions (Fig. 1). The E6 address is described by cortege e6; the type nwt describes E6 subnet/host address as a cortege consisting of the E6 address e6 and the mask mask. The packet pkt consists of E6 source (e6src) and destination (e6dst) addresses, the content data, the time stamp dt for the consequent QoS evaluating. The formats of the E6-RIP messages rtm and the routing table records rtr correspond to the descriptions listed in the Section 3. The type cha describes the information of the channel which can be either the data packet pk or the E6-RIP message rm; in more sophisticated versions special indicators of channels availability [5] can be employed. The type rt describes the routing table as the list of rtr records. The types buf, brtr describe the output buffers records for data packets and routing information correspondingly; the second field gives the destination port number. The type rtmin is employed for the internal temporary storing of the received routing message rtm together with the input port number ifn.

V. E6-RIP COMPONENTS

The E6-RIP protocol is represented by two components RIPprocess, RIPupdate shown in Fig. 4, 5 correspondingly. The component RIPprocess processes the received E6-RIP protocol messages while the component RIPupdate implements the updates forming and the timeouts processing.

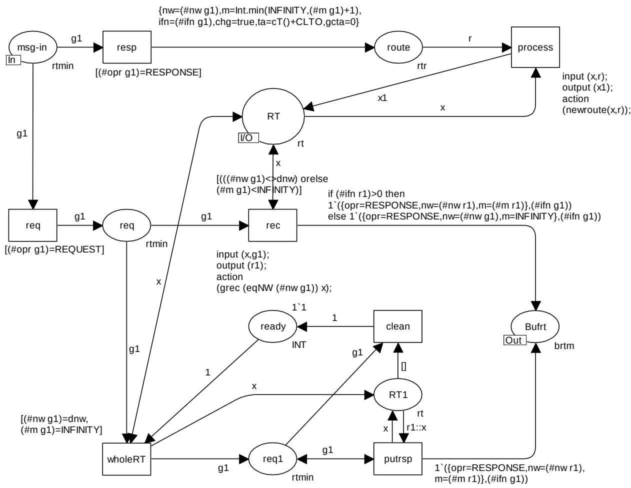


Figure 4. Component RIPprocess

The E6-RIP messages arrive into the place msg-in. The pair of transitions resp, req recognizes the RESPONSE and REQUEST operations correspondingly. At the request processing, there is considerable difference between the request of the whole routing table transmitting wholeRT and the request about a separate subnet (record) rec. At the response forming, the transition rec implements the search of the given record using the function grec and returns the corresponding record; at the record absence the metric INFINITY is indicated. For the transmitting of the whole routing table it is duplicated into the place RT1 and transmitted record by record into the request port via the transition putrsp; the transition clean implements the erasing of the empty table and returns the readiness indicator ready. The constant dnw corresponds to the default route.

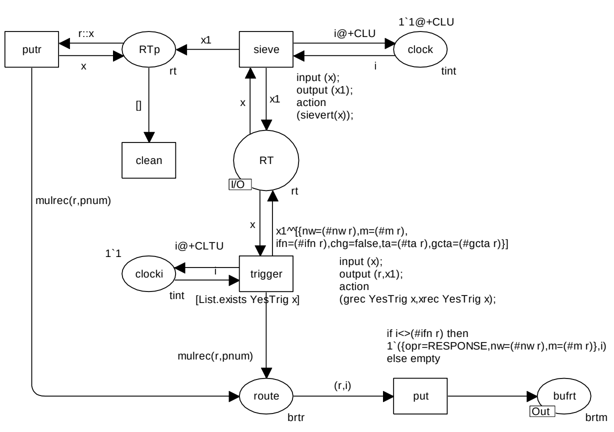


Figure 5. Component RIPupdate

At the response (route update) processing by the transition resp the message is transformed into the routing table format with the increment of metric, setting up the changes indicator chg and the ageing timeout ta; the record is stored into the place route. The processing of the route update is realized by the function newroute into the transition process inscriptions; the descriptions of the function newroute and the nested function of the routing table record updating procr are listed in Fig. 6.

The work of the component RIPupdate (Fig. 5) is initialized by the pair of timers: the regular updates timer clock and the trigger updates timer clocki. In the first case the whole routing table is retransmitted by the transition putr, in the second case – only one record (having the indicator chg set up) by the transition trigger. The copy of the whole routing table is saved into the place RTp for the further record by record retransmitting by the transition putr; the transition clean erases empty table. The function mulrec serves for the record propagation on all the ports. The updates are put into the place route from which they are transmitted into the buffer bufrt in the routing message format rtm by the transition put; the splitting of horizon is realized by the comparison with the source port number (i<>(#ifn r)).

```
fun procr(r0:rtr,r:rtr)=
if (#m r)=INFINITY andalso (#m r0)=INFINITY then r0
else
if (#m r)<INFINITY andalso (#m r0)<INFINITY andalso
  (#ifn r)<>(#ifn r0) andalso (#m r)>=(#m r0) then r0 else
if (#m r)=INFINITY andalso (#m r0)<INFINITY then
  {nw=(#nw r0),ifn=(#ifn r0), m=INFINITY,chg=true,ta=0,
gcta=cT()+CLGCTO} else
if (#m r)<INFINITY andalso (#m r)<(#m r0) then r else
if (#m r)<INFINITY andalso(#m r0)<INFINITY andalso
  (#ifn r)=(#ifn r0) andalso (#m r)=(#m r0) then
  {nw=(#nw r0),ifn=(#ifn r0),m=(#m r0),chg=(#chg
r0),ta=(#ta r),gcta=0} else r;
fun newroute([],r:rtr)=if (#m r)<INFINITY then [r] else [] |
newroute(r0::t,r:rtr)=if (#nw r0)=(#nw r) then
[procr(r0,r)]^^t else [r0]^^newroute(t,r);
fun sievert ([])=[] |
sievert(r::t)=if (#gcta r)>0 andalso (#gcta r)<=cT() then
sievert(t) else if (#ta r)>0 andalso (#ta r)<=cT() then
   ([{nw=(#nw r),m=INFINITY,ifn=(#ifn r),
chg=true,ta=0,gcta=cT()+CLGCTO}]^^sievert(t))
   else ([r]^^sievert(t));
fun mulrec (r,1) = 1`(r,1) | mulrec (r,i)= 1`(r,i)++mulrec(r,i-
1);
```

Figure 6. Routing information updating functions

At the routing table copying by the transition sieve, its filtrating according to the timeouts is realized by the function sievert (Fig. 6). An additional filtrating could be implemented into the trigger updates transition trigger for more thorough tracing the timeouts.

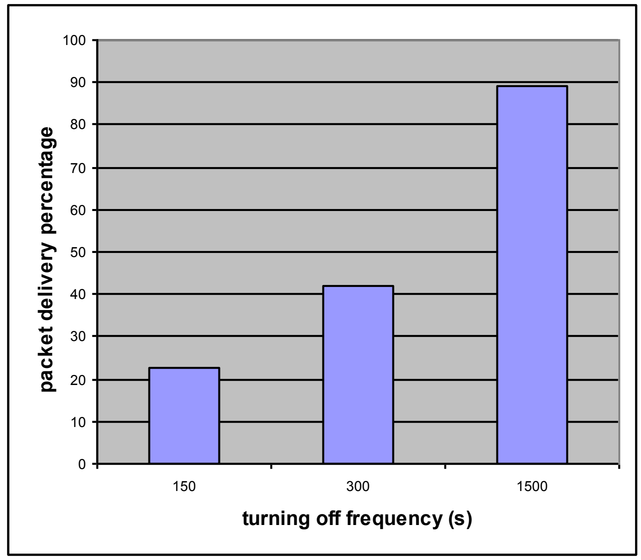
VI. ANALYSIS OF SIMULATION RESULTS

Developed components were used for the definite given E6 networks [1,6] modeling together with the terminal (subscriber) networks models for the traffic generating [6].

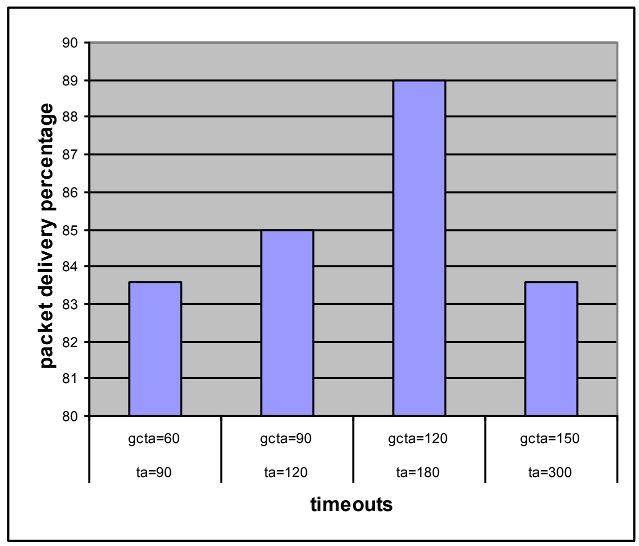
The measuring fragments were added to the terminal network model [6] for the evaluating of maximal and average packet delivery time and the performance [9]. On the networks with fixed structure, it was checked that the routing tables were filled up correctly. At the lagging traffic generating there is no lost packets. For the indirect evaluation of the protocol E6-RIP quality of work and its parameters choice, the number of lost packets was used. For the overload evaluation, the number of transmitted E6-RIP messages was calculated relatively to the useful traffic. Timed intervals up to one hour were simulated with the number of events (replications) about ten million; at the statistical data processing, confidence intervals of value 0,95 were applied.

Since the adaptation to fixed structure is rather trivial, the analysis was focused on the protocol behavior under the changes of the network structure. For this purpose the described models were supplied with the facilities of the separate ports and whole routers turning off as well as the separate calculating of the packets number lost as the result of the turning off. At the devices turning on, the requests on the whole routing tables transmitting from the neighbors were employed. The random uniformly distributed devices' turning off was realized with the period OffPer and the duration OffDur; the application of other turning off distribution laws is possible as well.

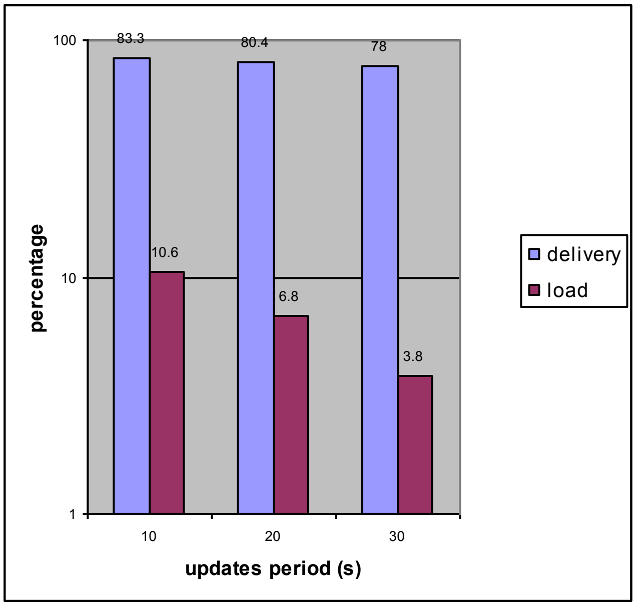
In Fig. 7 the results of the protocol parameters influence on the delivered packets percentage and the load created by the dynamic routing are represented; at the load evaluation in Fig. 7 c,d) the logarithmic scale was used. The duration of the turning off is set equal to 100-200 s. From Fig. 7 a) it is seen that the turning off period with the duration 200-400 s. (average 300 s.) is a critical one for the network functioning; the further evaluations were implemented for the period 1000-3000 s. (average 1500 s.). The standard updates timer period either increasing or decreasing lead to worst results regarding the standard timer (Fig. 7 b). The considerable influence on the delivered packets percentage exerts the regular updates period (Fig. 7c); but its decrease down to 10 s. increases the routing protocol load on the network up to 11%. The variant of the regular updates period about 20 s. looks as rather acceptable with the proportional decrease of the ageing and garbage collection timeouts to 120 and 80 s. correspondingly. The influence of the trigger updates period is rather insignificant (Fig. 7 d); its decrease lower than the standard value of 1 s. is not recommended. The poisoned reverse does not give considerable advantages over the simple splitting of horizon. The investigation of the value INFINITY influence was hampered in the current version of CPN Tools because it evokes the generating and processing the large scale network structures that requires considerable computational resources at modeling.



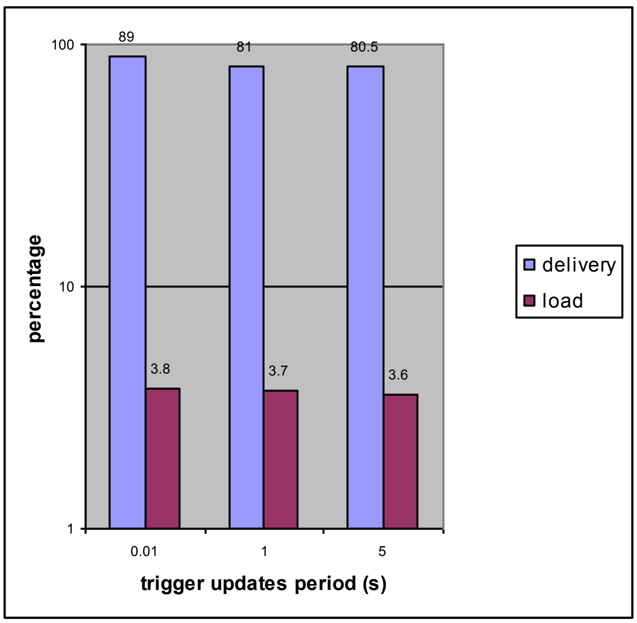
a) turning off frequency



b) timeouts



c) period of regular updates



d) period of trigger updates

Figure 7. The evaluation of the protocol parameters influence

VII. CONCLUSION

Thus, the principles of the known dynamic routing protocols adaptation into E6 networks were presented. The specifications and components for the protocol E6-RIP modeling were developed in the language of colored Petri nets.

The simulation of E6 networks series with turning off devices was implemented for the evaluation of the protocol robustness and its parameters choice. The protocol provides fast adaptation to the variable network structure.

REFERENCES

[1] Vorobiyenko P.P., Zaitsev D.A., Nechiporuk O.L. "World-wide network Ethernet?", Zviazok (Communications), no. 5, 2007, 14-19. In Russian.

[2] Vorobiyenko P.P., Zaitsev D.A., Guliaiev K.D. "Way of data transmission within network with substitution of network and transport layers by universal technology of data-link layer", Patent of Ukraine on utility model no. 35773, 2008. In Ukrainian.

[3] Fang L., Zhang R., Taylor M. "The Evolution of Carrier Ethernet Services – Requirements and Deployment Case Studies", Communications, vol. 46, no. 3, March 2008, 69-76.

[4] Guliaiev K.D., Zaitsev D.A., Litvin D.A., Radchenko E.V. "Simulating E6 Protocol Networks using CPN Tools", In Proc. of International Conference on IT Promotion in Asia, August 22-26, 2008, Tashkent (Uzbekistan), 203-208.

[5] Zaitsev D.A., Shmeleva T.R. Simulating Telecommunication Systems in CPN Tools: Students' book, Odessa: ONAT, 2006.

[6] Zaitsev D.A., Sakun A.L. "An Evaluation of MPLS Efficacy using Colored Petri Net Models", In Proc. of International Middle Eastern Multiconference on Simulation and Modelling (MESM'2008), Amman (Jordan), August 26-28, 2008, 31-36.

[7] Hedrick C. Routing Information Protocol, Network Working Group, RFC 1058, 1988.

[8] Zaitsev D.A., Shmeleva T.R. "Switched Ethernet Response Time Evaluation via Colored Petri Net Model", In Proc. of International Middle Eastern Multiconference on Simulation and Modelling, August 28-30, 2006. - Alexandria (Egypt), 2006, 68-77.

[9] Zaitsev D.A., Shmeleva T.R. "Evaluation of Ethernet Networks Characteristics using Parametric Petri Nets", Zviazok (Communications), no. 4, 2007, 62-67. In Russ.

Published: 9th International Workshop on Performability Modeling of Computer and Communication Systems (PMCCS-9), September 17-18, 2009, Eger (Hungary).