Investigation of the Fault Tolerance of the PIM-SM IP Multicast Routing Protocol for IPTV Purposes

Gábor Lencse, István Derka

Abstract—**IPTV services should use an IP multicast solution for a network bandwidth efficient delivery of the media contents. PIM-SM is the most commonly used IP multicast routing protocol in IPTV systems. A short introduction to the operation of PIM-SM is given. Its fault tolerance is examined by experimenting on a mesh topology multicast test network built up by XORP routers in a virtualized environment. Different fault scenarios are played and different parameters of PIM-SM and OSPF are examined if they influence and how they influence the outage time of an IPTV service. A formal model is given for the service outage time of the IPTV service on the basis of the results of the experiments.**

Index Terms—**IPTV, IP multicast protocols, PIM-SM, OSPF, fault tolerance, mesh networks, simulation models.**

I. Introduction

The global number of IPTV subscribers is growing rapidly [1]. As we have shown in [2], instead of IP unicast, an IP multicast solution should be used in IPTV systems that have a high number of active subscribers (except for the video-on-demand service). There were a number of IP multicast protocols invented, e.g. Distance Vector Multicast Routing Protocol (DVMRP, RFC 1075), Multicast Open Shortest Path First (MOSPF, RFC 1581), Core-Based Trees [3] (RFC 2189), Protocol Independent Multicast – Dense Mode (PIM-DM, RFC 3973) and Protocol Independent Multicast – Sparse Mode [4] (PIM-SM, RFC 4601). From these protocols, PIM-SM is the one that is commonly used in IPTV systems.

The probability of the failure of at least one element (e.g. router) of a network grows with the number of elements of the network. Large networks have redundant routers and transmission lines that are used for building alternate data pathes in case of failures. The multicast routing should also support this solution. For example, a fault tolerant solution for the Core-Based Trees was proposed in [5].

As for PIM-SM, the Rendezvous Point (RP, see explanation later) was identified as a single point of failure, as PIM-SM allows only one RP per multicast groups [6]. PIM version 2 introduced a standards-based mechanism for the RP fault tolerance and scalability using the Bootstrap Routers [7]. This mechanism makes possible for a multicast based IPTV system to survive the failure of the RP; however the switching over to the new RP is not always invisible for the customers, but may cause service outage for a certain amount of time. In our current research, we are interested in the length of the service outage time and the parameters it may depend on. Different scenarios were investigated and parameters were tested whether they have an influence on the length of the service outage time, and if so, how they influence it.

We expect that our results will be useful for both

1. the appropriate choice of the parameters of PIM-SM based multicast subsystems for IPTV systems and
2. building simulation models of the failure behaviour of the PIM-SM multicast protocol.

The second one is very important, because simulation (that is experimenting with a computerized model of the system) is a powerful tool for the performance and fault tolerance analysis of complex ICT (Information and Communication Technology) systems [8]; and measurement data gained with our experimenting with a real system is essential in the model building stage of a simulation project.

The remainder of this paper is organised as follows. First, an introduction to the operation of PIM-SM is presented (for more information see [9] or RFC 4601). Second, a very brief summary of IPTV is given. Third, the test environment is described. Fourth, the different kinds of experiments are presented and the results are interpreted. Fifth, formal models are given for the service outage time of the IPTV system in the function of certain parameters of PIM-SM and OSPF. Finally our conclusions are given.

II. The Operation of PIM-SM

Protocol Independent Multicast builds multicast trees on the basis of routing information obtained from a unicast routing protocol (e.g. RIP, OSPF) – this is why PIM is called “protocol independent”. It has four variants, from which the two most important ones are:

1. *PIM – Dense Mode* (RFC 3973) builds the multicast trees by flooding the whole network by multicast traffic and then pruning back the branches of the traffic distribution tree where no receivers of the multicast traffic are present.

2. *PIM – Sparse Mode* (RFC 4601) does not suppose group members everywhere thus sends multicast traffic into those directions where it has been requested using unidirectional *shared trees* rooted at the *Rendezvous Point*. It may optionally use shortest path trees per source.

In the rest of this paper, we use PIM-SM. PIM-SM does not have an own topology discovery method, but uses the Routing Information Base (RIB) of the unicast routing protocol applied in the *Autonomous System* (AS). With the help of this "outer" *Routing Information Base* (RIB), PIM-SM

Manuscript submitted Aug. 13, revised Dec. 31, 2012. This research was supported by the TÁMOP-4.2.2/B-10/1-2010-0010 project.

Gábor Lencse is with the Department of Telecommunications, Széchenyi István University, 9026 Győr, Egyetem tér 1, Hungary.
phone: +36-30-409-56-60, fax: +36-96-613-646, e-mail: lencse@sze.hu

István Derka is with the Department of Telecommunications, Széchenyi István University

builds its own *Multicast Routing Information Base* (MRIB). Unlike unicast RIB (that specifies the next router towards the destination of the packets) MRIB specifies the reverse path from the subnet to the router.

As PIM-SM is an *Any-Source Multicast* (ASM) protocol, the receivers need to find the source(s). The so-called *Rendezvous Point* (RP) is used for this purpose. The RP can be set statically by the administrator of the AS, or it can be elected from among the RP candidate routers.

There can be only one RP per multicast groups in the AS (or multicast domain) at a time. Note that there is a technique called *Anycast RP* (RFC 4610) that uses multiple instances of the RP in a single multicast domain using the same IP address (anycast addressing) and sharing their information about the sources with the Multicast Source Discovery Protocol (MSDP, RFC 3618). However, the failure of an instance of the RP still requires some kind of switching over to another instance and this switchover also causes outage in the IPTV service, so in this paper, we have chosen the clearer way of having one RP only and electing a new one if it fails.

The operation of PIM-SM has three phases. Now, we briefly describe what happens in these phases.

A. Phase One: RP-Tree

The *Rendezvous Point Tree* (RP-tree) is being built in the following way. The receivers send their *IGMP* (or MLD) *Join* messages with the required group address as destination IP address. The *Designated Router* (DR) of the receiver (that was elected from among the local routers before) receives the IGMP Join message and sends a *PIM Join* message to the RP of the required multicast group. This PIM Join message travels through the routers in the network and the visited routers prepare the appropriate MRIB entries thus the *RP-tree* is being built. The PIM Join messages have the marking: (*, G), where the first element is the IP address of the streaming source and the second one is the IP address of the multicast group. The star ("*") means that when a receiver joins a group, it will receive the traffic from all the sources that send steam to multicast group G. The PIM Join messages do not need to travel until the RP; it is enough to reach a point where the RP-tree has already been built. (The RP-tree is also called *shared tree* because the multicast traffic from all the sources uses the same tree.) The PIM Join messages are resent periodically while there is at least a single member in the group. When the last receiver of a leaf network leaves the group then DR sends a (*, G) *PIM Prune* message towards the RP so as to cut back the tree until the point where there are other active receivers connected.

When an S data source starts sending to a group, the first hop router (DR) of the source encapsulates the data packets of the source into unicast messages called *Register* messages and send them to the RP. The RP router knows from the Register messages that the source is ready to send the stream. The RP decapsulates the Register messages, and forwards the contained streaming data message to the appropriate multicast group (if it has at least a single member) using the RP-tree. The whole process is illustrated in Fig. 1.

Note that the multicasting is fully functional at end of phase one; the following two phases serve efficiency purposes only.

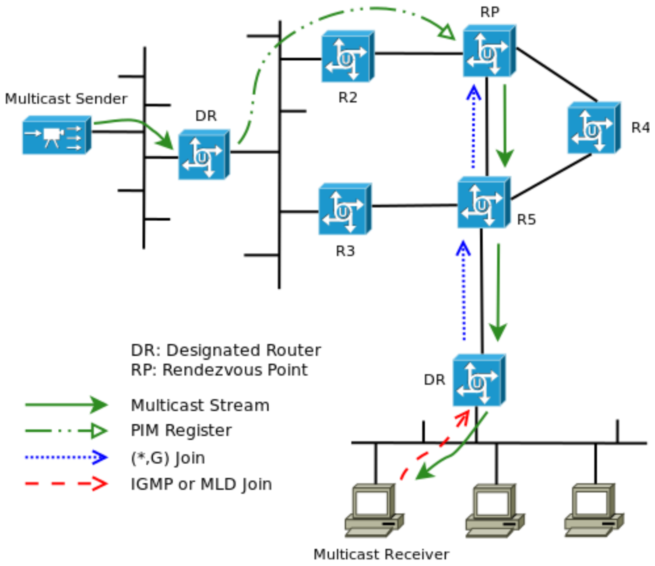

Fig. 1. The Operation of PIM-SM – Phase One

B. Phase Two: Register-Stop

The RP sends an (S, G) Join message to the source. As this message travels to the source, the routers along its path register the (S, G) pair to their MRIB (if they do not have it yet). When this Join message arrives to the subnet of the source (S) or to a router that already has an (S, G) pair registered in its MRIB, then the streaming data start flowing from the S source to the RP by multicast routing. Now, a *source-specific multicast tree* between the S source and the RP was built. After that, the RP sends a *Register-Stop* message to indicate that the first hop router of the source does not need to send Register messages (encapsulating the multicast data packets into unicast messages). Phase two is illustrated in Fig. 2.

C. Phase Three: Shortest-Path Tree

The path of the packets from the source to the receivers through the RP may be not optimal. To eliminate this, the DR of the receiver may initiate the building of a *source*

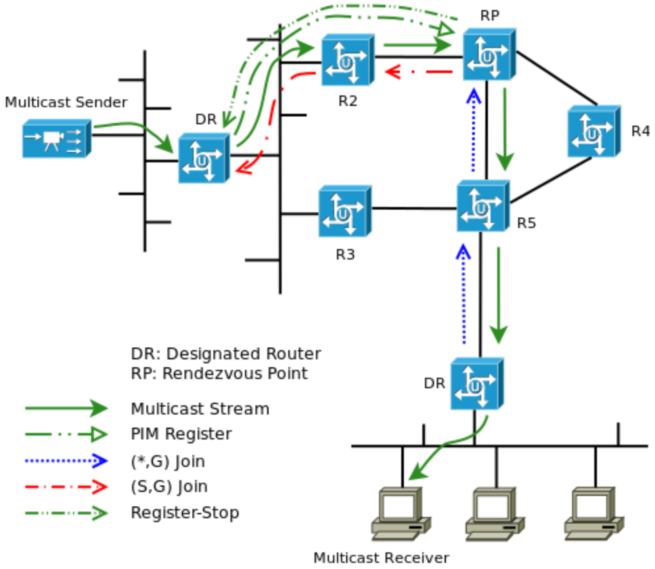

Fig. 2. The Operation of PIM-SM – Phase Two

specific shortest-path tree (SPT) towards the source (in this way possibly leaving out RP from the path). To do this, DR sends an (S, G) Join message to S. When this message arrives to the subnet of S or to a router that already has an (S, G) pair, then the streaming data start flowing from S to the receiver using this new SPT.

Now, the receiver receives all the streaming data packets twice. To eliminate this, the DR of the receiver sends an (S, G) Prune message towards the RP. (This is also known as an (S, G, rpt) Prune.) This message will prune the unnecessary tree parts and the streaming data will not arrive to the receiver through the RP-Tree any more. See Fig. 3.

D. The Built-in Fault Tolerance Mechanism of PIM-SM

It is an important element of the fault tolerance of PIM-SM that the RP does not need to be set up manually, it can be automatically elected from among those PIM-SM routers that were configured *Candidate RP* (C-RP).

The election uses the bootstrap mechanism described in RFC 5059. The *BSR router* is elected dynamically from among the PIM-SM routers that were configured *Candidate BSR* (C-BSR). All the C-BSR routers flood the multicast domain with their *Bootstrap messages* (BSM). The one with the higher priority wins. During the BSR election all the routers – including C-RP routers – learn the IP address of the BSR. After that, all the C-RP routers send their *Candidate-RP-Advertisement* (C-RP-Adv) messages to the BSR periodically. (The C-RP-Adv messages are sent in every *C_RP_Adv_Period* seconds, the default value is 60 seconds.) The BSR collects these messages, builds an *RP list* and advertises it also periodically for all the routers. The list is encapsulated into a BSM and is sent in every *BS_Period* seconds. All the routers – including the BSR and the C-RPs – can decide the *winner RP*[1] by the priority of the C-RPs. If the current RP fails to send its C-RP-Adv message to the BSR within *RP Holdtime* (a value included in the C-RP-Adv message) then BSR decides that it is dead and starts advertising the new RP list leaving out the dead one.

Notes:

1. RFC 5059 says that the RP candidate routers should set *RP Holdtime* to a value that is not less than 2.5*max{BS_Period, C_RP_Adv_Period} so that the system is able to tolerate the loss of some Bootstrap messages and/or C-RP-Adv messages.
2. The C-BSR routers also take care if the elected BSR fails, but that is not addressed in this paper.

E. The Choice of the Underlying Unicast Routing Protocol

As PIM-SM is *protocol independent*, there is a certain freedom in the choice of the underlying unicast routing protocol. The two most widely used protocols are the Routing Information Protocol (RIPv2, RFC 2453) and the Open Shortest Path First (OSPFv2, RFC 2328) for routing within a single autonomous system. Even though RIP is much simpler and more widely used in LANs than OSPF, it is not scalable and therefore it is not appropriate for the size of networks that are often used for providing IPTV services. This is why OSPF was chosen for our test network.

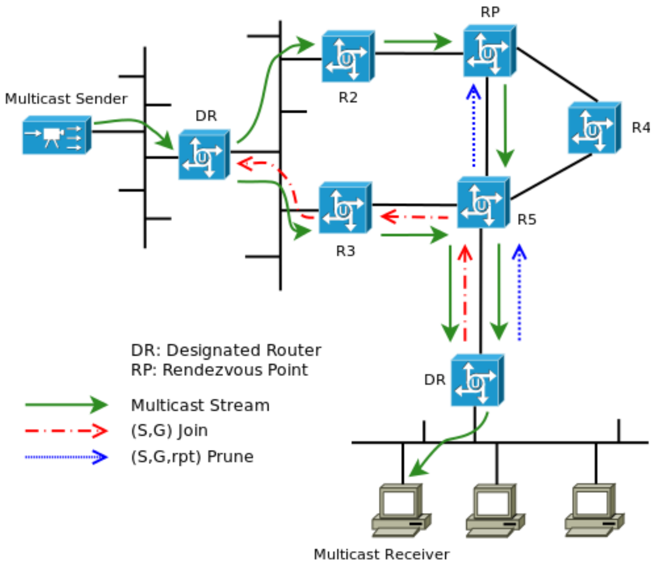

Fig. 3. The Operation of PIM-SM – Phase Three

Note that OSFP also uses a fault tolerance mechanism but it is much simpler than that of PIM-SM. The OSPF routers take care for their neighbours only. All the OSPF routers send *Hello* messages in every *Hello Interval* seconds to their neighbours. If they do not see a *Hello* message from a neighbour within the so called *Dead Interval* time they consider the given neighbour dead and they calculate new routes leaving out the dead neighbor.

III. IPTV IN A NUTSHELL

Nowadays, several data transmission technologies are available to transmit digital data (that may represent various media types, e.g. video, audio, text, etc. – the standard handles them in a uniform way) over different channels such as DVB-S/S2 via satellite, DVB-T/T2 via terrestrial, DVB-C/C2 via cable TV links and so on. In the TCP/IP based networks, the commonly used solution for delivering the digital video, audio and auxiliary data is based on the DVB-IPTV [10].

A general property of the above mentioned technologies is that they use the same MPEG2 Transport Stream (MPEG2-TS) format to organize the digital data (video, audio, etc) and additional service informations (SI/PSI tables) into a common frame. Basically, two types of MPEG2-TSs are available: the Single Program Transport Stream (SPTS), which includes only one service (e.g. TV program) and the Multiple Program Transport Stream (MPTS).

The MPEG2-TS (SPTS or MPTS) is divided into 188 bytes long packets (4 bytes header and 184 bytes data). In the IPTV environment, usually seven TS packets are embedded into one IP/UDP or IP/UDP/RTP packet and they are sent through the network. Unlike other DVB technologies, IPTV does not use broadcasting to deliver these packets. Instead, it uses IP multicast for the live or online streaming (e.g. live TV) and unicast for the offline services, for example VoD or Timeshift.

When a subscriber would like to watch the selected IPTV program his/her receiver joins to the TV program's preprogrammed IP multicast group. After the join process (a few seconds) the receiver will get continuously the

[1] There can be different RP-s for different multicast groups; RP-s are advertised together with the group address and netmask.

MPEG2-SPTS packets of the TV program through the IP multicast enabled network. If the subscriber switches over to another IPTV program then the receiver will leave the current one and join to the next IP multicast group.

IV. TEST ENVIRONMENT

In order to have a test network of reasonable size, a virtualization environment was used. The virtualization software was VmWare ESXi running on an IBM eServer BladeCenter LS20 using 5 blades having each 4GB RAM and two dual core AMD Opteron CPUs running at 2.2 GHz. The storage was mounted through NFS using Gigabit Ethernet network connection.

The topology of the test network was a mesh. It is not a typical topology for the commercial IPTV networks, but it was considered suitable for our experiments because it contains several redundant paths with equal costs. The mesh network contained 4 times 4 virtual routers interconnected by Layer 2 virtual switches. The virtual routers were built of virtual computers (1 virtual CPU, 512MB RAM, 10GB HDD) running Ubuntu 10.04 LTS operating system. The well known and widely used XORP [11] routing platform was chosen to implement both OSPF and PIM-SM for unicast and multicast routing, respectively. Two further virtual computers with the same configuration and operating system were added to the mesh network for the purposes of media streaming server and playing client. The VLC software of VideoLAN was used for both server and client purposes.

Note that our test system contained altogether 18 virtual computers and two Layer 2 virtual switches powered by 20 (5 times 4) CPU cores, thus each logical device had its own physical CPU and enough memory. Except for the moderate computing requirement of media streaming, all the other nodes had low computing requirements. In this way, it was ensured that the application of virtualization would not garble our results.

A. IP Configuration

Private IP addresses were used from the 192.168.0.0/16 network. The IP addresses of the virtual computers were configured manually as shown in Fig. 4. The network segments between two routers displayed by horizontal and vertical lines got IP addresses from 192.168.{1-12}.0/24 and 192.168.{13-24}.0/24 networks respectively. The last octets of the IP addresses of the interfaces are written next to the interfaces. The IP addresses of the network segments connecting the server and the client virtual computers are displayed in a similar manner.

B. OSPF Configuration

Because of the nature of the mesh, the OSPF protocol could be configured by the definition of peer-to-peer connections (it can be done if the neighbouring routers are interconnected by point-to-point links). A typical configuration fragment for an interface looks like follows:

```
ospf4 {
    router-id: 192.168.1.1
    area 1.1.1.1 {
        area-type: "normal"
        interface eth0 {
            link-type: "p2p"
            vif eth0 {
                address 192.168.1.1 {
                    interface-cost: 1
                    neighbor 192.168.1.2 {
                        router-id: 192.168.1.2
...
```

Configuring OSPF in this way made the network fully connected: unicast IP packets can be sent from anywhere to anywhere. Note that PIM-SM uses the unicast routing table (RIB) when building its own multicast routing table (MRIB).

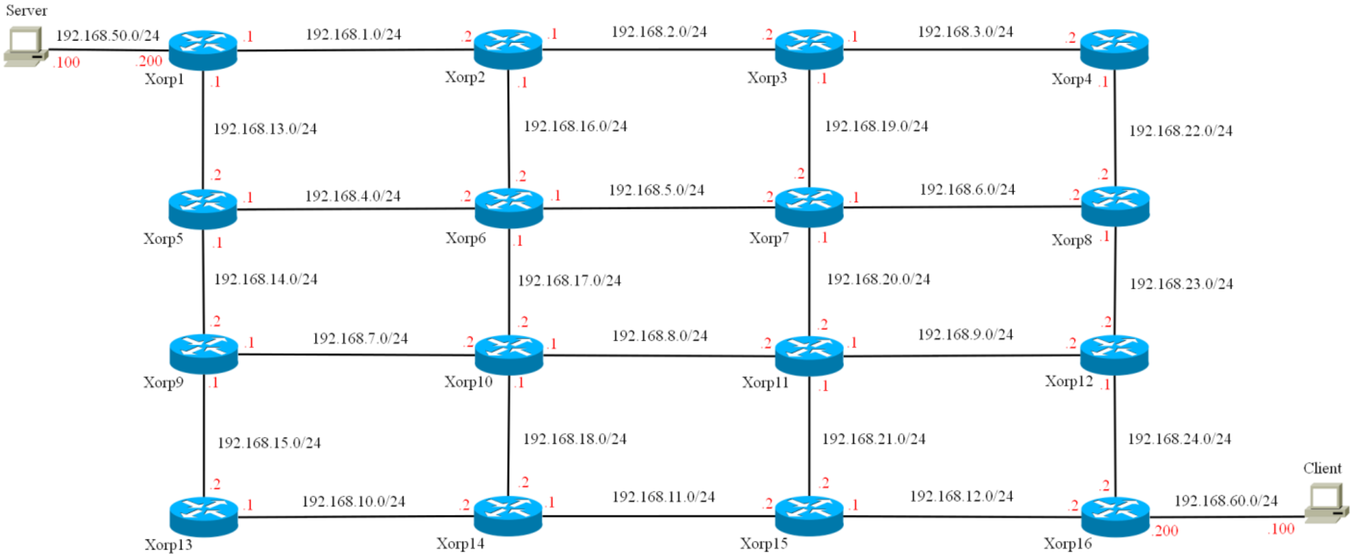

Fig. 4. Topology of the Test Network

C. PIM-SM Configuration

For PIM-SM, those and only those interfaces should be configured where PIM-SM has to handle multicast traffic. A typical configuration for an interface looks like follows:

```
pimsm4 {
    disable: false
    interface eth0 {
        vif eth0 {
            disable: false
            dr-priority: 101
            hello-period: 30
            hello-triggered-delay: 5
        }
    }
...
```

In order to be able to experiment with the fault tolerance of PIM-SM, the dynamic election of the RP was used. This required us to configure some routers as C-RP and at least one router as C-BSR. Routers **xorp2**, **xorp4** and **xorp14** were configured as both C-RP and C-BSR but with different priorities[2]. The **xorp2** router was the highest priority C-RP, the **xorp4** was the second highest priority one; **xorp14** was the highest priority C-BSR. A typical configuration for a router that was set as both C-RP and C-BSR looks like follows:

```
bootstrap {
    disable: false
    cand-bsr {
        scope-zone 224.0.0.0/4 {
            cand-bsr-by-vif-name: "eth0"
            bsr-priority: 102
        }
    }
    cand-rp {
        group-prefix 224.0.0.0/4 {
            cand-rp-by-vif-name: "eth0"
            rp-priority: 102
        }
    }
}
```

Considering the fact that in phase three there is no need for the RP, but a source-specific shortest path tree (SPT) is used for the transmission of the stream (that may not contain RP, or even if it contains RP then RP acts like a simple multicast router only), PIM-SM was configured so that it would never enter phase three. (The XORP implementation of PIM-SM gives three possible parameters for the control of switching to SPT, the simplest way is just to disable it.)

```
switch-to-spt-threshold {
    disable: true
    interval: 100
    bytes: 1024000
}
```

D. Time Synchronization

The important events of the measurements were logged into text files. In order be able to compare the timestamps of the events occurred on different virtual computers, the system times of the other virtual computers were synchronized to **xorp1** using the standard NTP protocol.

E. Streaming

A single program transport stream (SPTS) – that was demodulated and demultiplexed from a Hungarian DVB-T multiplex – was pre-recorded and used for all the measurements. The VLC server sent the stream to the 230.1.1.1 multicast IP group address using UDP. The VLC client received the stream and the standard **tcpdump** program was used to monitor (capture and record for offline analysis) the stream on the receiver side.

V. Experiments and Results

A. Testing the Failure of the RP

Hypothesis 1: Killing the RP on **xorp2** router will stop the stream for a while, but the stream will be restored when a new RP is elected. The length of the service outage time likely depends on how much time elapsed from the last Candidate-RP-Advertisement (C-RP-Adv) message when the RP is killed.

The measurements were controlled by a script executed on **xorp2** router. This script did the following: after starting XORP and the streaming, it made sure that **xorp2** is the actual RP. Then it waited until XORP sent a C-RP-Adv message. From that time it waited until a *predefined delay* (it was a parameter, see later). After that it started the **measure.sh** scripts both on the DR of the server and on the DR of the client (these scripts recorded the IP address of the actual RP in every second) and sent a marker (ICMP echo request) to the client and killed the RP.

The *predefined delay* was increased from 5 seconds to 55 seconds in 5 seconds steps. (As C-RP-Adv is sent in every 60 seconds by the defaults settings of XORP, there would be no point in increasing the delay above 55 seconds.) The whole measurement was executed 11 times.

The results of the measurements can be found in Fig. 5. Here, and also in all the following figures the standard deviation is displayed by a vertical section: the Y coordinates of the two ends of the section are set as:

(average - std. dev, average + std. dev.)

The service outage time values justify hypothesis 1: even though they show large fluctuations, there is a visible tendency that a larger delay from the last C-RP-Adv usually results in shorter service outage time.

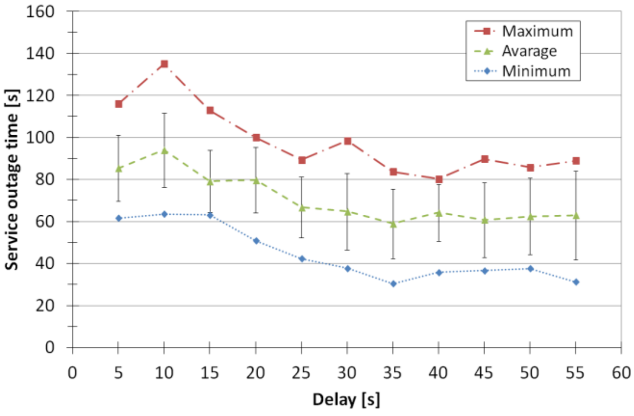

Fig. 5. Service Outage Times in the Function of the Delay from the Last ***C-RP-Adv Message*** *to the Stopping of the* ***RP*** *on xorp2 Router*

[2] As it is commonly used in Unix like systems, the lower numeric value means higher priority.

B. Testing the Failure of the Complete PIM-SM router

Hypothesis 2: Switching off the operation of the complete XORP on **xorp2** router will stop the stream for a while, but the stream will be restored when the underlying unicast routing (OSPF) finds new route (that does not contain **xorp2** router) from the DR of the server to the DR of the client.

This process will result in shorter service outage time as the default timeout values of OSPF are shorter than that ones of PIM-SM. We also expect that the length of the outage time will show no correlation with the time elapsed from the last C-RP-Adv message when the RP is killed.

The measurements were taken in a similar way than before. The results can be found in Fig 6. They justify hypothesis 2: the service outage times are much shorter and they show no correlation with the time elapsed from the last C-RP-Adv message. Both prove that *no new RP is necessary for the restoration of the stream*.

Hypothesis 3: The length of service outage time caused by the switching off the operation of the complete XORP on **xorp2** router depends on the time elapsed from the last Hello message of the OSPF protocol.

The default values of the OSPF *Hello Interval* and *Dead Interval* are 10 seconds and 40 seconds respectively. For testing purposes, the first one was raised to 35 seconds and similar series of the measurements were performed in the way that the delay from the last OSPF *Hello message* before the stopping of XORP was increased from 5 seconds to 30 seconds in 5 seconds steps. The results can be found in Fig. 7. They justify hypothesis 3: the average service outage times are very close to the time that was left from the *Dead Interval* of OSPF at the time of stopping XORP. (The stream was restored because OSPF calculated a new route that did not contain the **xorp2** router.)

C. Limiting the service outage time by parameter tuning

As we have shown in section B, if the service outage was caused by the complete failure of a multicast routing node[3] which is an element of the path from the DR of the server to the DR of the client then the service outage time was determined by the parameters of the underlying unicast routing protocol. In our experiments, the service outage time was upper bounded by the *Dead Interval* of OSPF. The actual value of the service outage time depended on the elapsed time from the last OSPF *Hello* message at the time of the failure of XORP.

Hypothesis 4: The service outage time caused by the complete failure of a XORP router can be limited by an appropriate setting of the OSPF *Dead Interval* parameter.

The measurements were taken in the usual way but using 20 seconds and 15 seconds as OSPF *Dead Interval* and *Hello Interval*, respectively. The values of delay from the last OSPF *Hello* message to the failure the XORP were 5 and 10 seconds. The results can be found in Table 1. They justify hypothesis 4.

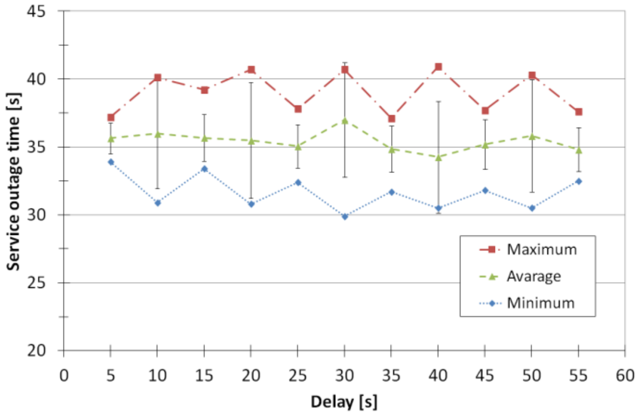

Fig. 6. Service Outage Times in the Function of the Delay from the Last ***C-RP-Adv Message*** *to the Stopping of* ***XORP*** *on xorp2 Router*

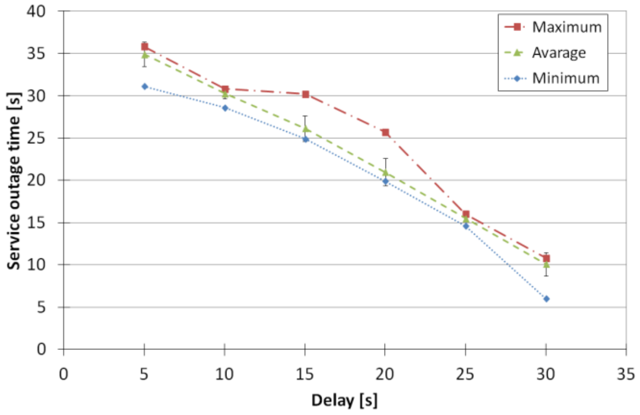

Fig. 7. Service Outage Times in the Function of the Delay from the Last ***OSPF Hello Message*** *to the Stopping of XORP on xorp2 Router*

TABLE 1. SERVICE OUTAGE TIMES IN THE FUNCTION OF THE DELAY FROM THE LAST ***OSPF HELLO MESSAGE*** *TO THE STOPPING OF XORP ON* **XORP2** *ROUTER* ***USING 20 SECONDS OSPF DEAD INTERVAL***

Delay [s]	Service outage time [s]			
	min	max	average	std. dev.
5	14,8	15,8	15,45	0,39
10	9,8	10,8	10,45	0,38

The significance of the findings of hypotheses 4 is that the time of the service outage caused by the complete failure of a multicast node can be limited by the appropriate choice of the *Dead Interval* parameter of OSPF. Note that the service outage time cannot be arbitrarily decreased in this way for at least two reasons:

1. The choice of the *Dead Interval* parameter of OSPF has a consequence on the frequency of the OSPF Hello messages. This frequency should not be too high as these messages consume both network and router capacity.

2. The exchange of the topology information and the recreation of the routing tables in OSPF require a certain amount of time. Though this time was negligible in our experiments due to the small size of our test network, the situation can be different in the case of a real life multicast network for IPTV.

[3] It can be the RP, but it is not necessarily the RP.

Finding a similar way of limiting the service outage time caused by the failure of the RP only would be a natural idea, however it is much more difficult.

Hypothesis 5: The service outage time caused by the failure of the RP only (but XORP remained working) can not be efficiently limited by the choice of the PIM-SM *RP Holdtime* parameter.

In the fifth series of measurement, the value of the PIM-SM *RP Holdtime* parameter was changed from its default value of 150 seconds to 75 seconds. As the XORP platform does not provide a method for modifying the value of C_RP_Adv_Period, its value was left 60 seconds. Note that these settings do not comply with RFC 5059 (see Note 1. in section II.D. of this paper), but in our small test network the loss of C_RP_Adv or BSM messages is not a serious issue.

The results of the measurements can be found in Fig. 8. They justify hypothesis 5: the service outage time values are not significantly smaller than in Fig. 5 and the results show similar fluctuations as they did it in Fig. 5.

The reason of this behaviour and especially of the fluctuations can be found in the operation of the fault tolerance mechanism of PIM-SM. Two independent and unsynchronised timers are used for measuring C_RP_Adv_Period and BS_Period in the C-RP routers and in the Bootstrap router, respectively. If the killing of the RP is synchronized to one of them, the unpredictable value of the other one causes "random" fluctuations in the service outage time.

Hypothesis 6: Repeating the first series of measurements of killing the RP on `xorp2` router but measuring the delay from the last BSM message received (instead of from the last C-RP-Adv message) when the RP is killed will produce similar results that is the longer delays result in shortest service outage time – at least in tendency –, but there will be similar fluctuations.

The results in Fig. 9 justify hypothesis 6: the average service outage times show a decreasing tendency in the function of the delay from the last BSM, but they are not monotonous and the measured values show similar fluctuations as it could be seen in Fig. 5.

The findings of hypotheses 5 and 6 deserve some discussion. These results do not give us a straight forward way of limiting the service outage time in the case if the RP fails. However they give us an important lesson: it is worth entering the third phase of PIM-SM not only for efficiency reasons (that is using SPT for faster delivery) but also for achieving shorter service outage time in case of the failure of a multicast router due to the faster recovery of OSPF (see hypotheses 3 and 4).

Note that even though the failure of the RP could be easily simulated for experimenting purposes using the `xorpsh` interface of the XORP routing platform; in practice, the complete failure of a router is much more typical than the failure of its RP functionality only.

VI. Towards a Formal Model for the Service Outage Time

As the simulation of large and complex systems may require a huge amount of memory and processing power, the models used for simulation should contain only the details that are relevant for the purposes of the simulation [12].

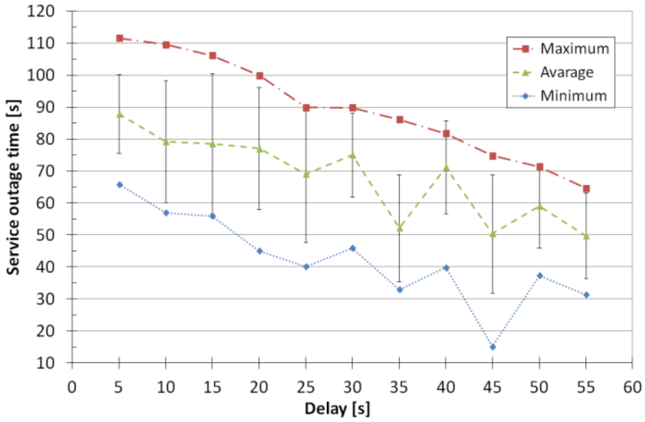

Fig. 8. Service Outage Times in the Function of the Delay from the Last ***C-RP-Adv Message*** *to the Stopping of the RP on xorp2 Router* ***using 75 Seconds PIM-SM RP Holdtime***

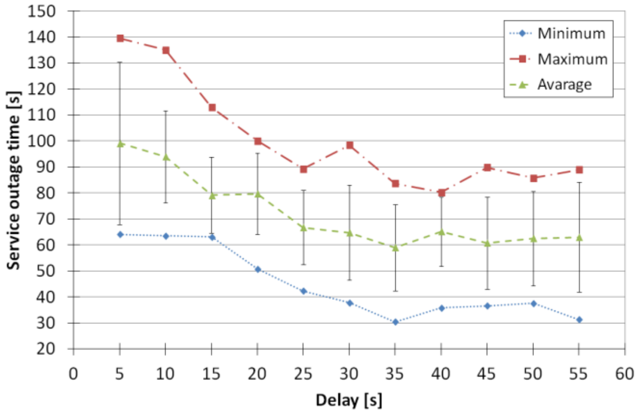

Fig. 9. Service Outage Times in the Function of the Delay from the ***Time of the Last BSM Received*** *to the Stopping of the* ***RP*** *on* ***xorp2*** *Router*

For example, when the focus in placed on certain networking applications, the detailed behaviour of the lower layer components of the network (e.g. the 1-persistent CSMA/CD protocol played by Ethernet NICs at MAC layer) are usually omitted or if they have an important influence on the communication they are modelled by a much simpler phenomenon (e.g. the packet loss due to collisions is modelled by random drop of packets with a fixed or traffic volume dependent packet loss ratio).

The service outage time caused by the complete failure of a XORP router can be modelled as follows:

$t_{SO} = ODI - DLH + t_{D\&R}$

Where t_{SO}, ODI, DLH and $t_{D\&R}$ denote the time of service outage, the length of OSPF *Dead Interval*, the delay from the last OSPF *Hello* message at the time of the failure of the XORP router and the time OSPF uses for the distribution of topology information and for the recalculation of the routes, respectively. Note that $t_{D\&R}$ is not negligible in general, as it depends from the size of the network. In a practical simulation model for a given fix size of network, $t_{D\&R}$ can be approximated by a constant.

As for the two other values, ODI is a constant parameter of OSPF, and DLH can be modelled by random variable that takes its values from [0, OSPF *Hello Interval*] according to uniform distribution.

The service outage time caused by the failure of the RP only could be formally modelled as follows:

$$t_{SO} = PRH - f_1(DLC) - f_2(DLB) + t_{NRP}$$

Where t_{SO}, PRH, DLC, DLB and t_{NRP} denote the time of service outage, PIM-SM *RP Holdtime*, the delay from the last PIM-SM C-RP-Adv message of the RP at the time of the failure of the RP, the delay from the last PIM-SM BSM received at the time of the failure of the RP and the time necessary for the designated router of the source and of the server for switching over to the new RP, respectively. The $f_1(.)$ and $f_2(.)$ functions are necessary because in spite of the fluctuations it can be clearly seen in Fig. 5 and Fig. 10 that the service outage time was decreased by less than the value of DLC and DLB. They could be estimated, however because of the before mentioned rare nature of the failure of the RP only the estimation probably does not worth the effort.

VII. Directions of Future Research

The formal model for the service outage time presented above should be validated. This is planned to be done among the next year PhD research tasks of the second author of this paper.

VIII. Conclusions

The operation and the fault tolerance mechanism of PIM-SM were introduced. A mesh topology test network of virtual computers running the XORP router software was built. In different series of experiments, the service outage time of an IPTV system was measured in the function of different parameters of PIM-SM and OSPF in the second phase of the PIM-SM protocol.

It was shown that in case of the complete failure of the RP or any router in the path of the multicast stream from the DR of the server to the DR of the client the service outage time depends on the OSPF Dead Interval parameter and the delay elapsed from the last OSPF Hello message at the time of the failure. A formal model was also given for the service outage time. It was also shown that in the much less common case of the failure of the RP functionality only (not the failure of the complete router that serves actually as the RP) the service outage time depends on a number of different factors and it cannot be easily limited by parameter tuning due to the unpredictable conditions of two unsynchronised timers. A formal model was also given for the service outage time in this case.

We conclude that it is worth switching to the third phase of PIM-SM for fault tolerance considerations, because in that phase the service outage time is shorter, predictable and can be limited by the appropriate selection of the Dead Interval parameter of OSPF.

IX. References

[1] International Television Expert Group, "Global IPTV market (2009-2013)" http://www.international-television.org/tv_market_data/global-iptv-forecast-2009-2013.html

[2] G. Lencse and B. Steierlein, "Quality of service and quality of experience measurements on IP multicast based IPTV systems", Acta Technica Jaurinensis, Vol. 5, No. 1, pp. 55-66. 2012.

[3] A. J. Ballardie, P. F. Francis, and J. Crowcroft, "Core Based Trees", ACM SIGCOMM Computer Communication Review Vol. 23, No. 4, pp. 85–95. August 1993. DOI:10.1145/167954.166246.

[4] S. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. Liu, L. Wei, "The PIM architecture for wide-area multicast routing", IEEE/ACM Transactions on Networking, Vol. 4, No. 2, pp. 153-162. April 1996. DOI:10.1.1.39.7251

[5] Weijia Jia, Wei Zhao, Dong Xuan, Gaochao Xu: "An efficient fault-tolerant multicast routing protocol with core-based tree techniques", IEEE Transactions on Parallel and Distributed Systems, Vol. 10, No. 10, pp. 984-1000. 1999. DOI:10.1.1.74.157

[6] M. Sola, M. Ohta and T. Maeno: "Scalability of internet multicast protocols", in: Proc. of INET'98, Geneva, Switzerland, July 1998.

[7] Silvano Da Ros, Content Networking Fundamentals, Cisco Press, 2006, ISBN: 1-58705-240-7

[8] L. Muka and G. Muka, "Creating and using key network-performance indicators to support the design and change of enterprise infocommunication infrastucture", in: Proc. of 2012 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS 2012), Genoa, Italy, July 8-11, 2012, Volume 44, Books 12, ISBN: 978-1-61839-982-3, pp. 737-742.

[9] Beau Williamson, Developing IP multicast networks, Volume 1, Cisco Press, 2000, Indianapolis, IN, USA. ISBN: 1-57870-077-9

[10] Digital Video Broadcasting (DVB); Transport of MPEG-2 TS Based DVB Services over IP Based Networks, ETSI TS 102 034 V1.4.1 (2009-08)

[11] XORP Inc. and individual contributors, XORP user manual, Version 1.8-CT, 2010.

[12] G. Lencse and L. Muka, "Managing the resolution of simulation models", in: Proc. of the 2008 European Simulation and Modelling Conference (ESM'2008), Le Havre, France, Oct. 27-29, 2008, EUROSIS-ETI, ISBN: 978-90-77381-44-1, pp. 38-42.

Gábor Lencse received his MSc in electrical engineering and computer systems at the Technical University of Budapest in 1994, and his PhD in 2001. He has been working for the Department of Telecommunications, Széchenyi István University in Győr since 1997. He teaches Computer networks, Compurer architectures, IP based telecommunication systems and the Linux operating system. Now, he is an Associate Professor. He is responsible for the specialization of the information and communication technology of the BSc level electrical engineering education. He is a founding member of the Multidisciplinary Doctoral School of Engineering Sciences, Széchenyi István University. The area of his research includes discrete-event simulation methodology and performance analysis of computer networks. Dr. Lencse has been working part time for the Department of Telecommunications, Budapest University of Technology and Economics (the former Technical University of Budapest) since 2005. There he teaches Computer architectures and Media communication networks.

István Derka received his Msc at Faculty of Electrical Engineering and Informatics at the Technical University of Budapest in 1995. He worked for the Department of Informatics from 1999 to 2003 and since then has been working for Department of Telecommunications, Széchenyi István University in Győr. He teaches Programming of communication systems and Interactive TV systems. He is an Assistant Professor. The area of his research includes multicast routing protocols and IPTV services in large scale networks.