TOWARDS THE MODELLING OF THE FAULT TOLERANCE MECHANISM OF THE PIM-SM MULTICAST ROUTING PROTOCOL IN AN IPTV ENVIRONMENT

Gábor Lencse and István Derka
Department of Telecommunications
Széchenyi István University
H-9026 Győr,
Hungary
E-mail: lencse@sze.hu

KEYWORDS
IP multicast protocols, PIM-SM, OSPF, fault tolerance models, mesh networks, simulation models, IPTV.

ABSTRACT

The fault tolerance mechanism of the PIM-SM IP multicast routing protocol is investigated in order to be able to model it. The analysis is done by playing different fault scenarios on a mesh topology multicast test network built up by XORP routers in a virtualized environment. Different parameters of the PIM-SM and the OSPF protocols are examined if they influence and how they influence the outage time of an IPTV service. The results of the experiments provide important factors for building a formal model of the service outage time of an IPTV service.

INTRODUCTION

Recently, IPTV is a hot research topic. An IP multicast solution should be used in IPTV systems that have a high number of active subscribers (Lencse and Steierlein 2012). 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 (Ballardie et al. 1993) (RFC 2189), Protocol Independent Multicast – Dense Mode (PIM-DM, RFC 3973) and Protocol Independent Multicast – Sparse Mode (Deering et al. 1996) (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. a 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 paths 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 (Jia et al. 1999). 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 (Sola et al. 1998). PIM-SM version 2 introduced a standards-based mechanism for RP fault tolerance and scalability using the Bootstrap Routers (Ros 2006). 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 in building simulation models of the failure behaviour of the PIM-SM IP multicast protocol. Having a good model is very important, because simulation is a powerful tool for the performance and fault tolerance analysis of complex ICT (Information and Communication Technology) systems (Muka and Muka 2012); and our measurement results may help in building a good simulation model.

The remainder of this paper is organised as follows. First, a brief introduction to PIM-SM is given. For more information see (Williamson 2000) or RFC 4601. Second, the test environment is described. Third, the different kinds of experiments are presented and the results are interpreted. Fourth, 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.

PIM-SM IN A NUTSHELL

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 our research focuses on *PIM – Sparse Mode* (RFC 4601) only. PIM-SM 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. 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 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 RP in a single domain using the same IP address (anycast addressing) and sharing the their information about the sources with the Multicast Source Discovery Protocol (MSDP, RFC 3618). However, the failure of an instance of RP still requires some kind of switching over to another instance, 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 the following three phases:

1. Building a *Rendezvous Point Tree* from the receivers to the RP and the registration of the sources
2. Building *Shortest Path Tree* (SPT) from the RP to the source and Register-stop
3. Building the SPT from the receivers to the source

Now, we briefly describe what happens in these three phases.

Phase One: RP-Tree + the Registration of the Sources

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 *RP-tree* is being built. The PIM Join messages have the marking: (*, G). 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 steams 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 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 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 sends them to the RP. The RP router learns from the Register messages that the source is ready to send the stream. 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.

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

Phase Two: Building SPT from RP to S + Register-Stop

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 table (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 table, then the streaming data flow from the S source to RP by multicast routing. Now the *Shortest Path Tree* (SPT) between S and RP was built. After that, RP sends a *Register-Stop* message to indicate that the first hop router of the source does not need to send Register messages.

Phase Three: Building SPT from the Receivers to S

The path of the packets from the source to the receivers through the RP may be suboptimal. To eliminate this, the DR of the receiver may initiate the building of a *source specific shortest-path-tree* (SPT) towards the source. To do this, the 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 starts 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 message will prune the unnecessary tree parts and the streaming data will not arrive to the receiver through the RPT any more.

The Built-in Fault Tolerance Mechanism of PIM-SM

It is an important element of the fault tolerance of PIM-SM that 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 message*s (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. BSR collects these messages, builds an *RP list* and advertises it also periodically for all routers. The list is encapsulated into a BSM and it is sent in every *BS_Period* seconds. All the routers – including BSR, and C-RPs – can decide the *winner RP* by the priority of the C-RPs. If the current RP fails to send its C-RP-Adv message to the BSR within the *RP Holdtime* (its value is included in the C-RP-Adv message) then BSR decides that the RP is dead and starts advertising the new RP list leaving out the dead one.

Notes:

1. RFC 5059 says that RP candidates 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.

The Choice of the Underlying Unicast Routing Protocol

As PIM-SM is *protocol independent*, there is certain a 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.

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.

THE 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 network containing 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 (Xorp 2010) 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.

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. 1. 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.

In order to be able to experiment with the fault tolerance of PIM-SM, the dynamic election of 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. The **xorp2** router was the highest priority C-RP, **xorp4** was the second highest priority one; **xorp14** was the highest priority C-BSR.

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 the RP), PIM-SM was configured so that it would never enter phase three.

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.

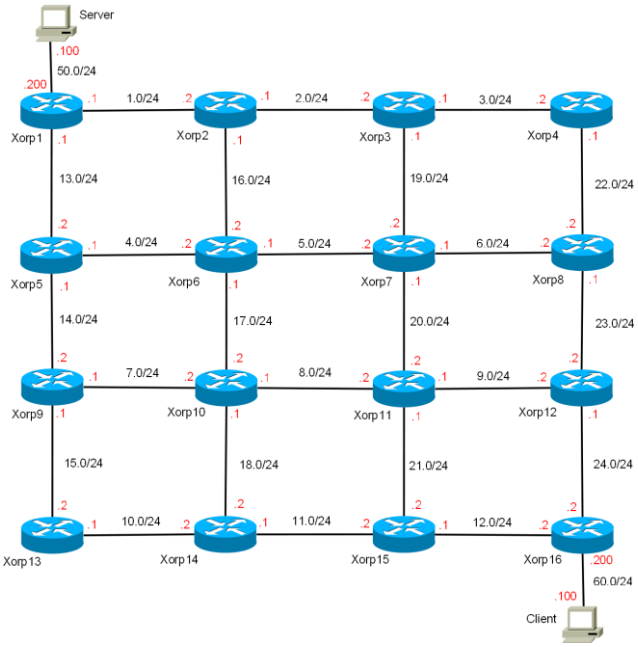

Figure 1: Topology of the Test Network (IP addresses are from the 192.168.0.0/16 network)

EXPERIMENTS AND RESULTS

Series of experiments were executed to test if a given parameter of PIM-SM or OSPF influences the service outage time of the streaming service and if yes how those parameters influences it. All the measurements were controlled by scripts. What was common in them is that they started the media streaming, waited until a specified PIM-SM or OSPF message arrived, that waited until a *predefined delay* that was a parameter taking its values from a given range, then stopped a given functionality of PIM-SM or OSPF (causing the failure of the streaming), and then measured the time elapsed until the restoration of the stream. All the measurements were repeated 11 times; average and standard deviation were calculated.

Testing the Failure of the RP

After receiving a PIM-SM Candidate-RP-Advertisement (C-RP-Adv) message and a *predefined delay* elapsed, the RP of PIM-SM was killed on the **xorp2** router. The *predefined delay* was increased from 5 seconds to 55 seconds in 5 seconds steps. (As C-RP-Adv is done in every 60 seconds by the default settings of XORP, there would be no point in increasing the delay above 55 seconds.)

Killing the RP on the **xorp2** router stopped the stream for a while, but the stream was restored when a new RP was elected. The length of the service outage time depends on how much time elapsed from the last C-RP-Adv message when the RP was killed. The results of the measurements can be seen in Fig. 2. Even though they show fluctuations, there is a visible tendency that a larger delay from the last C-RP-Adv usually results in shorter service outage time. But the fact that the service outage time is not a monotonous function of the delay from the last C-RP-Adv message suggests that the service outage time is probably depends on some other parameter(s) too.

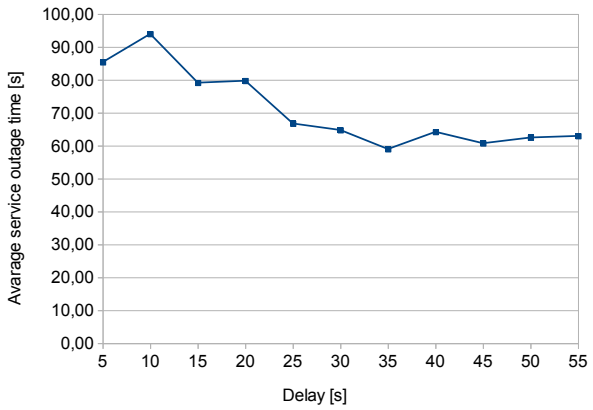

Figure 2: Service Outage Times in the Function of the Delay from the Last PIM-SM C-RP-Adv Message to the Stopping of the RP

Our second series of measurements were similar to the first series with the difference that the delay was measured from the last PIM-SM Bootstrap Message (BSM) received before RP was killed. The results are presented in Fig. 3. 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 2.

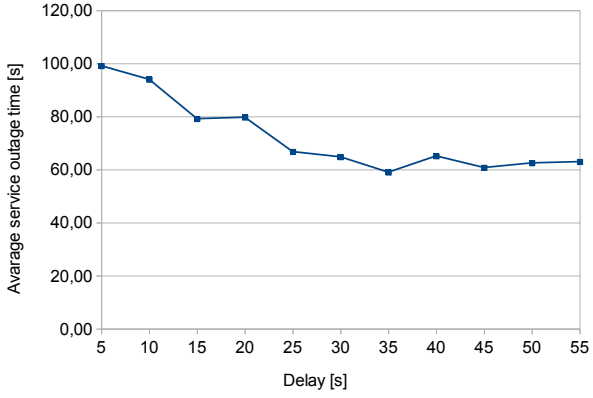

Figure 3: Service Outage Times in the Function of the Delay from the Last PIM-SM BSM Message to the Stopping of the RP

Testing the Failure of the Complete PIM-SM router

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 for this series of experiments.

After receiving an OSPF *Hello message* and a *predefined delay* elapsed, the complete XORP on the **xorp2** router was stopped. The *predefined delay* was increased from 5 seconds to 30 seconds in 5 seconds steps. Stopping the complete XORP on the **xorp2** router meant the stopping of the OSPF functionality of the router causing the failure of the stream. The stream was restored when the neighbouring OSPF routers detected that the **xorp2** router was dead and calculated a new route. The results in Fig. 4 show that the service outage time depends on the delay from the last OSPF *Hello message* to the stopping of OSPF and it is much shorter than it was in the first series of experiments. The latter one also proves that *no new RP was necessary for the restoration of the stream*.

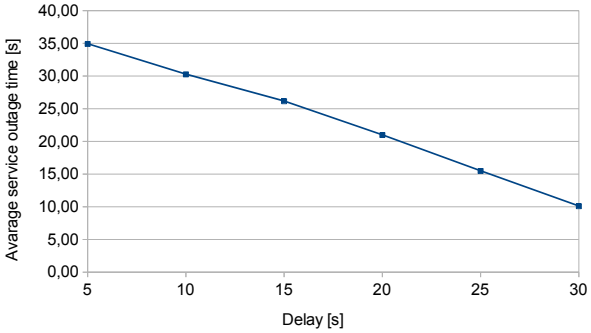

Figure 4: Service Outage Times in the Function of the Delay from the Last OSPF Hello Message to the Stopping of the XORP

Limiting the service outage time by parameter tuning

As we have shown recently, if the service outage was caused by the complete failure of a multicast routing node (it can be the RP, but it is not necessarily the RP) 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.

Now, we show that the service outage time caused by the complete failure of a multicast node can be limited by an appropriate setting of the OSPF *Dead Interval* parameter. The measurements were taken in the same way as in the previous series of measurements but using 20 seconds and 15 seconds as OSPF *Dead Interval* and *Hello Interval*, respectively. The values of the 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.

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. Further series of measurements were performed investigating the effect of the value of PIM-SM *RP Holdtime* parameter but the results were not convincing. (Space permits no more detailed discussion here.)

Table 1. Service Outage Times in the Function of the Delay from the Last OSPF Hello Message to the Stopping of XORP using 20 Seconds OSPF Dead Interval

Delay (seconds)	*Service Outage Time (seconds)*			
	min	*max*	*min*	*std. dev.*
5	*14,8*	*15,8*	*15,45*	*0,39*
10	*9,8*	*10,8*	*10,45*	*0,38*

The fact that the service outage time can be limited by the value of OSPF *Dead Interval* gives 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 as the recovery of OSPF is faster than the recovery of PIM-SM.

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.

TOWARDS BUILDING A FORMAL MODEL OF THE SERVICE OUTAGE TIME

To some up the findings above, the service outage time caused by the complete failure of a multicast node is proportional with the OSPF Dead Interval. The value of the delay from the last OSPF Hello Message to the failure of the node directly decreases the value of the service outage time.

Note that our test network was very small in size, thus the time necessary for the distribution of the topology information and for the recalculation of the routes was negligible, however in a real life size network they are to be taken into consideration.

The service outage time caused by the failure of the RP only was found much harder to grasp. It seems that both the delay from the last C-RP-Adv message and the delay from the last BSM are in a negative correlation with the service outage time. It is very likely that fluctuations were seen in the case of the measurements triggered by each of them were caused by the influence of the other one. The two kinds of PIM-SM messages (C-RP-Adv and BSM) are not synchronised to each other or to a common signal thus if the value of one of the two delays were selected the other one was unpredictable in our experiments.

CONCLUSIONS

We have shown that in the case of the complete failure of any router in the path of the multicast stream, 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.

We have also shown that in the much less common case of the failure of the RP functionality only, the service outage time depends on both the delay from the last C-RP-Adv message and the delay from the last BSM but we could not give an exact model due to the unpredictable conditions of the two unsynchronised messages.

Our results provide important factors for building a formal model of the service outage time of an IPTV service.

ACKNOWLEDGEMENT

This research and publication was supported by the TÁMOP-4.2.2/B-10/1-2010-0010 project.

REFERENCES

Ballardie, A. J.; P. F. Francis and J. Crowcroft. 1993. “Core Based Trees”, *ACM SIGCOMM Computer Communication Review* Vol. 23, No. 4, pp. 85–95.

Deering, S.; D. Estrin; D. Farinacci; V. Jacobson; C. Liu and L. Wei. 1996. “The PIM architecture for wide-area multicast routing”, *IEEE/ACM Transactions on Networking*, Vol. 4, No. 2, pp. 153-162. April 1996.

Jia, W.; W. Zhao; D. Xuan; G. Xu. 1999. “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.

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

Muka, L. and G. Muka. 2012. “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) Volume 44, Books 12, pp. 737-742.

Ros, S. D. 2006. *Content Networking Fundamentals*, Cisco Press, ISBN: 1-58705-240-7

Sola, M; M. Ohta and T. Maeno. 1998. “Scalability of internet multicast protocols”, in: *Proc. of INET'98*, Geneva, Switzerland, (July).

Williamson, B. 2000. *Developing IP multicast networks*, Volume 1, Cisco Press, 2000, Indianapolis, IN, USA.

Xorp Inc. and individual contributors. 2010. *XORP user manual*, Version 1.8-CT

AUTHOR BIOGRAPHIES

GÁBOR LENCSE was born in Győr, Hungary. He received his MSc in electrical engineering and computer systems at the Technical University of Budapest in 1994, and his PhD in 2001. Dr. Lencse has been working for the Department of Telecommunications, Széchenyi István University in Győr since 1997. He teaches Computer networks, Networking protocols and the Linux operating system. Now, he is an Associate Professor. The area of his research includes discrete-event simulation methodology and performance analysis of computer networks.

ISTVÁN DERKA was born in Győr, Hungary. He 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.