Best Effort Hierarchical Aggregation Tree for IPTV Signaling

Radim Burget, Jakub Muller, Dan Komosny

Faculty of Electrical Engineering and Communication, Brno University of Technology, Czech Republic

Summary
For the kind of applications such as IPTV a support of interaction between content provider and subscribers is a big challenge for the coming years. This paper extends the hierarchical aggregation method for determining as structure of hierarchical tree in more practical manner and describes TTP protocol, which should realize session communication.
Key words:
IPTV, Multimedia communication feedback, Communication system software, Hierarchical aggregation

1. Introduction

As more and more households are connected with broadband internet connection a new type of services are emerging. One of these services is also the IPTV service. According to current research the TV development is divided into four stages [8]. The first and four stages are in the name of interaction of subscribers with media content provider. However, the Real-time Transport Protocol and especially Real-time Control protocol is not prepared for such kind of services and for bigger number of receivers connected in session would lead either to a higher bandwidth consumption or significantly longer time for feedback signaling aggregation.

To cope with this problem several optimization methods have emerged such as filtering, biasing. However all these methods neglect some portion of receiver signaling. The hierarchical aggregation [1] seems to be quite promising method because it does not neglect any signaling, is very fast and highly scalable. A lot of work has been published about hierarchical aggregation however all the papers assume unlimited number of so-called feedback targets, which of course does not reflect a real conditions.

This paper summarizes a way how the ideal hierarchical aggregation tree structure should be constructed. At the same time it proposes a way how to construct HA tree with a given number of feedback targets. In the next chapter the tree transmission protocol is described and updated. Finally, the results are concluded.

Manuscript received November 5, 2008
Manuscript revised November 20, 2008

2. Mathematical background

2.1 Ideal conditions

As mentioned earlier the quality of service measurement is realized through sending sender reports (SR) and receiver reports (RR). The bandwidth consumed by these messages should use just 5% of the total bandwidth reserved for this service (see Equation 3) as defined in RTP/RTCP [2] standard. Therefore the reports are sent in the periods, which can be determined by Equation 1 for SR reports and Equation 2 for RR reports. n stands for number of receivers, PL_{RR} stands for packet length of RR message, similarly PL_{SR} stands for packet length of SR message. Using these equations the sending of messages is being spread over time and the session can scale to any numbers of receivers.

However as the period depends on the number of receivers linearly, it can lead to such a big periods that the obtained values by sender are quite old and do not reflect the current state in the session. For instance let we assume session with 10^5 receivers and 1Mbps bandwidth. This would lead approximately to T_{RR} equal to 60 minutes.

$$T_{\mathrm{SR}} = \frac{1 \cdot PL_{\mathrm{SR}}}{25\% \cdot BW_{\mathrm{RTCP}}} \quad [s] \tag{1}$$

$$T_{\mathrm{RR}} = \frac{n_{\mathrm{R}} \cdot PL_{\mathrm{RR}}}{75\% \cdot BW_{\mathrm{RTCP}}} \quad [s] \tag{2}$$

$$BW_{\mathrm{RTCP}} = 5\% \cdot BW \quad [bit \cdot s^{-1}] \tag{3}$$

The idea behind hierarchical aggregation is that the reception of RR messages is spread over several so-called feedback targets (FT) (see Equation 4), where n_{FT} stands for a number of the FT in the lowest level of the tree. FT is a new member type in the session and its only function is to for tree structure and gathers RR messages from receivers, aggregate the obtained messages to a single packet, which is a sort of histogram and retransmits the packet to one of the FT in the higher level of hierarchical tree. The FT in the higher level again aggregates all the received histograms into a new message and sends it to one of the FT in higher level of the HA tree. This is repeated until all the messages reach the root of this tree, so-called root feedback target, and thus the information about the whole session is collected on the single place.

$$T_{\mathrm{RR}} = \frac{\mathrm{PL}_{\mathrm{RR}} \cdot {}^{n}/_{n_{\mathrm{FT}}}}{75\% \cdot BW_{\mathrm{RTCP}}} \quad [s] \tag{4}$$

To avoid sending the messages too frequently, especially in case where there is few receivers when the broadcasting starts, we suggest to limit the minimal report period T_{RR}, T_{SR} to 5 seconds. This constant is also proposed in RTP/RTCP [2] standard. Thus until the number of receivers reaches some number (see Equation 5) it will not affect resulting period length.

$$N_5 = \frac{5\mathrm{sec} \cdot 75\% \cdot BW_{\mathrm{RTCP}}}{\mathrm{PL}_{\mathrm{RR}}} \quad [-] \tag{5}$$

Now let's take a look how the hierarchical tree can be built. For the simplicity reason let's first assume ideal conditions, where the unlimited number of FTs is available. When we know the number of receivers *n*, we can determine the required height of the tree (see Equation 6) and according to the Equation 7 we can determine how much FTs should be in each level of the hierarchical tree.

$$H_{\mathrm{FT}}(n) = \begin{cases} 1 & \text{if } n \le N_5 \\ 1 + H_{\mathrm{FT}}({}^{n}/_{N_5}) & \text{otherwise} \end{cases} \quad [-] \tag{6}$$

$$L_{\mathrm{FT}}(n) = \lceil {}^{n}/_{N_5} \rceil \quad [-] \tag{7}$$

Similarly the total number of required feedback targets can be determined according to Equation 8.

$$N_{\mathrm{FT}}(n) = \begin{cases} 1 & \text{if } n \le 0 \\ L_{\mathrm{FT}}(n) + N_{\mathrm{FT}}({}^{n}/_{N_5}) & \text{otherwise} \end{cases} \quad [-] \tag{8}$$

Thus we can construct ideal hierarchical tree, which will transmit the feedback from huge number of receivers in a really short time. Everything works well till a sufficient number of feedback targets is there. However, once you do not have the sufficient number of FTS, the Equation 8 cannot be used. In the next section constructing of the ideal tree with a given number of feedback targets is discussed.

2.2 Best Effort Hierarchical Tree

As mentioned earlier, when the required number of feedback targets (see Equation 8) is greater than the actual number of available feedback targets, the equations 6, 7, 8 are useless. In this section the case with a given number of receivers is discussed.

The Equation 9 express total time of feedback propagation from receivers to the root feedback target according to the number of feedback targets in the first, second and up to $H_{FT}(n)^{th}$ layer. Let's the height of the tree, which can be determined by equation 6, is denoted as *h*. The sum of the time through all the layers represents the total time of propagation.

$$T(L_{\mathrm{FT}}{}^{(h-1)}, L_{\mathrm{FT}}{}^{(h-2)}, \cdots, L_{\mathrm{FT}}{}^{(1)}) = \frac{L_{\mathrm{FT}} \cdot \mathrm{PL}_{\mathrm{RR}}}{{}^{(h)} \cdot \mathrm{BW}_{\mathrm{RR}}} + \sum_{i=1}^{(h)} \frac{L_{\mathrm{FT}}{}^{(i-1)} \cdot PL_{\mathrm{RR}}}{L_{\mathrm{FT}}{}^{(i)} \cdot BW_{\mathrm{RR}}} \quad [s] \tag{9}$$

This equation is general and can be applied for any height of the hierarchical tree. However, the practically usable number of receivers is only in orders of millions or up to tens of millions of receivers. Therefore we can limit the maximal height of the tree by up to three levels.

According to the Equation 9 we can express the total time of the signaling propagation in the single level tree by Equation 2, with two levels by Equation 10 and for three levels by Equation 11.

$$\frac{\partial T(L_{\mathrm{FT}}{}^{(1)})}{\partial L_{\mathrm{FT}}{}^{(1)}} = 0 \tag{10}$$

$$\frac{\partial T(L_{\mathrm{FT}}{}^{(2)}, L_{\mathrm{FT}}{}^{(1)})}{\partial L_{\mathrm{FT}}{}^{(1)} \partial L_{\mathrm{FT}}{}^{(2)}} = 0 \tag{11}$$

Unfortunately, the solution of the equation 11 will lead to polynomial of fourth order and the analytical expression of the equation roots would be quite complex. As depicted in Figure 1, the dependency of resulting time on number of FTs in the second tree layer is continuous and smooth and therefore some optimization method will give very good results.

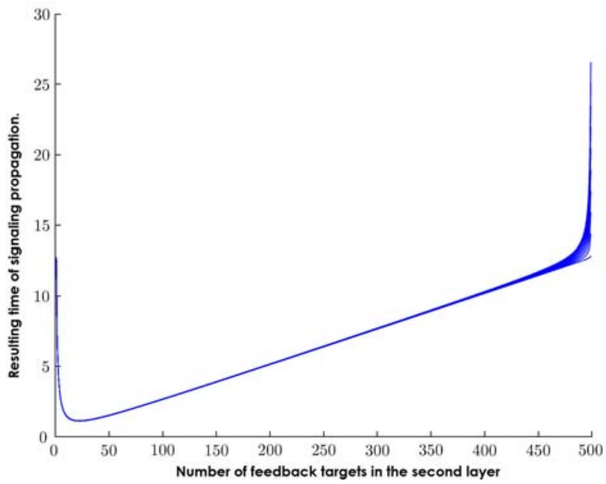

Figure 1 - Resulting time of propagation according to the

The resulting tree is then chosen according to a minimal value of the total time of feedback propagation.

2.3 Placement of Feedback Targets

To determine the placement of feedback targets there was defined so-called Tree Transmission Feedback algorithm [1]. Roughly speaking, the feedback targets can by selected by some kind of clustering algorithm and the clusterization should be performed iteratively level by level. Another approach can be also selecting the feedback targets randomly. However, in this case considerable bandwidth can be wasted.

$L_{FT}^{(0)}$ stands for layer where the root feedback targets is. Thus the number of the FTs in this layer must equal to one.

3. Tree transmission protocol

In the session there are four types of members: sender, receiver, feedback target and feedback targets manager. All the types represent applications, which are deployed on some machine in network and there is also possibility to deploy several of them on a single machine. However due to the performance issues there is recommended that each has its own machine and especially feedback targets are not only separated to different hardware but they are also distributed in the network.

3.1 Feedback Targets Initialization

The first thing, which must be performed before the session will start collecting receiver signaling feedback is to register all the feedback targets (FT) to the feedback target manager (FTM) and to determine the relative feedback target positions. Therefore when the feedback target, is started it just sends registration packet to the FTM. On the basis of this message the FTM reply with measurement requests, which contains several other randomly chosen feedback target IP addresses. This measurement obtains so-called round trip delay time (RTT) to other feedback targets. It is performed via ICMP protocol with so-called echo request packet and echo reply packet. The delay between sending the request and obtaining the reply stands for this RTT value. The measured values are then transmitted back to the FTM.

When the FTM obtains most of these values it can determine network position of these hosts. This relative position can be used further for optimal FTs selection and for optimized tree construction (see Figure 2).

3.2 Tree initialization

Before the start of IPTV streaming the sender has to request the FTM for the feedback tree, so that the signaling from receivers could be gathered and collected by the created hierarchical feedback tree. The number of receivers usually can be estimated for the next session, for example from streaming history statistics. Therefore the sender can request FTM for assigning some hierarchical tree for a particular number of senders. Thus the tree can be proposed for a particular number of receivers. FTM selects set of FTs according to this request, forms them logically to a tree structure and replies to the sender with the unique tree identification number. As this number is unique, not only one sender can utilize this set of FTs but several senders can share their capacity. When the streaming starts, the receiver is aware of the set of the feedback targets and sends reports to the particular FT about e.g. quality of service, votes for some polling etc. (see Figure 3).

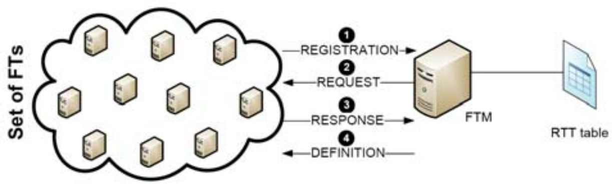

Figure 2 - Feedback targets registration

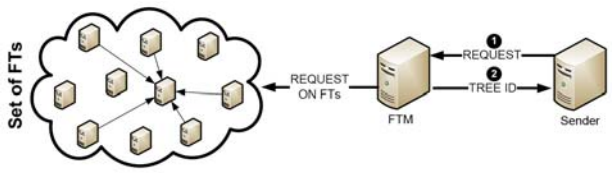

Figure 3 - Obtaining of hierarchical tree ID

4. Tree in network

After the FTM calculates all the necessary values (see Equations 5, 6, 8, 7), it starts initializing the tree in the network. For this purpose the definition packet is used. It sends the packet to the specific IP address of registered and chosen FT. There are two possible situations. The FTM can receive information packet with confirmation. That means that FT agrees with all parameters. Or FTM can receive information packet with declining and than it has to contact another FT.

4.1 Adding FT into the tree

Adding FT into the tree has several steps. You can see these steps in Figure 4. The first step is initializing FTs with definition packet. Definition packet contains information not only about level of FT but also about number of members etc. Level of the tree defines if the FT will summarize data from receivers or from FTs into the RSI [3] packets. In the second step, each FT sends information packet if it agrees with parameters or not. The third step represents monitoring multicast channel and reading information, which is sent periodically and contain actually state of tree generated and managed by FTM. Each FT is able to find IP address of next FT based on data contained in this information multicast channel.

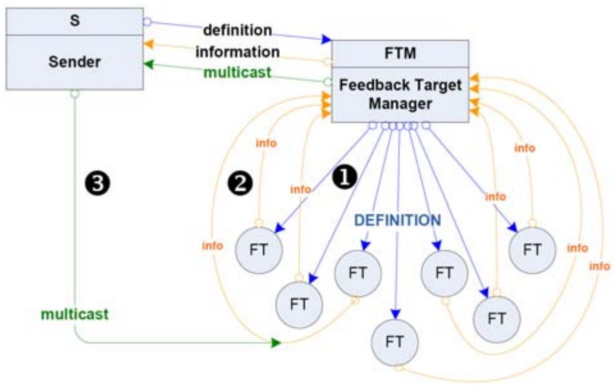

Figure 4 - Creation of the tree

4.2 Removing FT from the tree

We also need to remove certain FTs from tree from time to time. Removing FT from the tree does the FTM. If it decides to remove FT from tree, it sends information packet with this request. FT has no other options than agree. It works for some time until all its members stops sending their data to this FT. All members find out that this FT works no more, because the FTM doesn't generate information about it in information multicast channel no more. The client can be theoretically even mobile. He can move from place to place and look for the closest FT. If it finds out that another FT is closer it easily changes the target where the data are transferred. New FT has to indentify if all it parameters are fine. If not, the FT sends this information to FTM and it recalculates the structure of the whole tree or only parameters in some areas of the tree.

4.3 Receivers

Receivers do not care about the situation in the feedback channel. We can say that FT in the lowest level represents the target destination for receivers in its group. Receivers think that they are sending data directly to the sender. They also monitor the information multicast channel and detect how many members the particular FT can handle. Receiver calculates the periodical interval using this number. This is the main reason why they can send their data so frequently. The minimum interval is generally defined to 5s. The RTP/RTCP protocol [2] uses the same limit. The reason for this is to generate as much information as sender needs for a quality signaling. There is no need to generate data more often because they will bring no more use. Receivers can join the session in any time and therefore the structure of the tree can be changed also at any time. So, receivers have to monitor the information multicast channel and change the target if they find the better (”closer”) one.

4.4 Dealing with unexpected situations

We never know how many clients will be connected to the session. This is why the TTP protocol must be dynamic to adapt any requirement of the hierarchical tree. Each FT is activated only by the FTM and the number of members for each FT is also calculated by it. We want to have a stable type of the tree structure and therefore the FTM adds a randomization for this number which allows changing the number of members but not the type of the tree (count of FTs in the tree). Before this number reaches the upper or lower limit, FT informs FTM about this situation and FTM must deal with it. It can change the tree either with adding new FT in the tree, remove one FT or only defines new parameters for all set of FTs. The problem can be also on the FT side. The FT can fail and stop sending its RSI [3] messages to the higher level of the feedback tree. But FT in the higher level notices this problem and informs the FTM about the issue. The FTM responds with a tree change again. It tries to activate a new FT situated near to the position of the failed one or , in the worst scenario, recalculates the whole tree. The main idea of this process is that all members must have the same information about the tree. If anything happens somewhere in the tree and it cannot be resolved locally, the whole structure of the tree must be updated.

4.4 Usage of TTP and future extension

As we said and described earlier, TTP gives us opportunity to transfer huge amount of data via the narrow channel (small bandwidth), which we want to use. TTP protocol can be used in real-time applications like e.g. IPTV, where we can have even up to millions of clients and standard RTCP feedback channels, where communication cannot be fast enough for such a big number of clients. Changing the way of the collecting signalization from client is the solution and it can be performed by TTP protocol. On the other hand, we are using a powerful hardware in case of IPTV, but we cannot use such capabilities for example in area of sensor networks. This new branch of communication networks is growing very rapidly in the last years and many research teams are focused in it. There we cannot count with powerful servers of nodes but only with cheap devices (sensors). We would like to modify the TTP so it can be used also in this area. The first test seems to be quite promising.

5. Conclusion and future direction

Hierarchical aggregation seems to be quite a promising technology for large-scale topologies with a single source. As the receivers and feedback targets are organized hierarchically, there must be a way to organize them effectively. In this paper the construction of a hierarchical tree is proposed and the idea of integration of hierarchical aggregation with coordinating systems is described.

The further directions of this work can focus e.g. on speed optimization of propagation over hierarchical structure, combining advantages of coordinate systems Netvigator and GNP, design of more sophisticated algorithms for clustering, which will take into account also non-uniform spread of receivers or investigate capabilities of hierarchical tree and its resilience to structure changing or optimizing the process of receiver coordinates localization. A space for optimizations is also in selection of FTs according to the results from clustering algorithm, which is used in the TTF algorithm.

Acknowledgments

This work was supported by the Academy of Sciences of the Czech Republic, project 1ET301710508.

References

[1] Vit Novotny and Dan Komosny. Optimization of large-scale RTCP feedback reporting in fixed and mobile networks. In ICWMC '07: Proceedings of the Third International Conference on Wireless and Mobile Communications, page 85, Washington, DC, USA, 2007. IEEE Computer Society.

[2] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RFC 3550: RTP: A Transport Protocol for Real-Time Applications. Technical report, IETF, 2003.

[3] J. Ott, J. Chesterfield, and E. Schooler. RTCP Extensions for Single-Source Multicast Sessions with Unicast Feedback. Technical report, IETF, 2004.

[4] DSL Home-Technical Working Group. Cpe wan management protocol, 2004.

[5] Radim Burget Dan Komosny. Ttf simulation. Technical report. [online], Multicast IPTV Research Group, 2008, http://adela.utko.feec.vutbr.cz/projects/tree-organizing.html.

[6] Wendi Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient Communication Protocols for Wireless Microsensor Networks. In International Conference on System Sciences, Maui, HI, January 2000.

[7] Frank Dabek, Russ Cox, Frans Kaashoek, and Robert Morris. Vivaldi: a decentralized network coordinate system. SIGCOMM Comput. Commun. Rev., 34(4):15–26, October 2004.

[8] Harkirat Singh, ChangYeul Kwon, Chiu Ngo, "IPTV over Wireless LAN: Promises and Challenges",IEEE Consumer Communications and Networking Conference, Las Vegas, NV, Jan 10-12, 2008.

Ing. Radim Burget, *(1982) received his M.Sc. degree in Faculty of Information Technology in 2006 and he is the PhD student and assistant at the Department of Telecommunications of Brno University of Technology in Czech Republic. He is engaged in research focused on IPTV. He has experience with development of J2EE applications.

Ing. Jakub Müller, *(1984) received his M.Sc. degree in Electrical Engineering in 2008 and he is the PhD student and assistant at the Department of Telecommunications of Brno University of Technology in Czech Republic. He is engaged in research focused on IPTV, hierarchical aggregation and mobile application. He has experience with development of J2EE applications.

Ing. Dan Komosný, Ph.D., *(1976) received his Ph.D. degree in Teleinformatics in 2003. In 2003 he became an assistant professor at the Department of Telecommunications, Brno University of Technology. His research is aimed on multimedia systems; communication protocols; real-time communication; VoIP technology; IPTV technology, IP network quality of service and communication pragmatics for Wireless Sensor Network. He has practical experience with visual software development; real-time software development and formal description of communication protocols. He received or participated in several projects focused on networking research. He is a member of IEEE Communication Society.