Large-Scale RTCP Feedback Optimization

Vít Novotný, Dan Komosný
Dept. of Telecommunications
Faculty of Electrical Engineering and Communications
Brno University of Technology
Brno, Czech Republic
novotnyv@feec.vutbr.cz, komosny@feec.vutbr.cz

Abstract— Nowadays, multimedia streaming is a common service on the Internet including both fixed and mobile networks. The RTP protocol is usually used to transfer the media. The RTCP protocol is an accompanying protocol for the RTP and it is used for feedback transmission in order to control multimedia session behaviour. The issue is the bandwidth dedicated for the RTCP protocol. As defined in the RTP specification, it is limited to 5% of the total allowed bandwidth. This is a limiting factor for large-scale media streaming services based on Source-Specific Multicast (SSM) since the RTCP bandwidth is shared among all the receivers. It causes large delays in sending feedback data from each receiver. For the purpose of solving this issue, a hierarchical structure of receivers with summarisation nodes has been proposed. The paper introduces an optimization of this hierarchical structure in order to achieve the lowest feedback transmission intervals. The proposal is supported by a simulation in Matlab environment. Furthermore, the paper deals with a design of a new protocol called Tree Transmission Protocol (TTP) that was proposed to organize session nodes into a hierarchical tree structure.

Index Terms—streaming services, RTP, RTCP, ASM, SSM, hierarchical feedback,

I. INTRODUCTION

The RTP and RTCP protocols are used for media transmission and feedback, respectively, in multimedia streaming services in an IP environment (fixed or mobile) (RFC 3550 - [1] and RFC 3551 - [2]). The original standard [1] was designed for the Any Source Multicast (ASM) architecture, for example for videoconferences. Multicast channels were specified for data transmission of both the RTP and RTCP protocols. At present, applications like video, sound or data streaming over the Internet are quite common and it is sure that their number will increase. A good example is the IPTV service. This and the other similar services are based on the one media streaming server – many clients architecture, i.e. the Source Specific Multicast (SSM) [3], [4] (see Figure 1 for fixed Internet and Figure 2 for mobile subscribers). Because of one media streaming server and many receivers it is not useful to keep a full-mesh topology for multicast routing in both directions. Only the server needs a complete feedback for monitoring the quality of service from its clients, therefore the unicast communication in the direction from the clients to the server is a better solution [5]. However, the clients need to know at least the number of clients to determine the frequency of the Receiver Report (RR) RTCP packets transmission. Several methods were designed. One simple method called 'Simple Feedback Model' ([6], [7]) relies on the plain reflection of the client RRs received over unicast channels back to the clients over multicast channels. Though simple, this solution has some disadvantages, because a lot of overhead data is generated. The second method is based upon the aggregation of the RR information in the streaming server (usually) received from the clients, in the assembly of a summarization packet while removing irrelevant data, and in transmitting this summary report back to the clients. To enable this, an extension of the original RTCP specification in the form of an Extend Report (XR) packet had to be adopted [6].

Figure 1. An example of SSM media streaming on the Internet

Figure 2. Media streaming to mobile networks

The XR RTCP summarization packet consists of a header and blocks that contain, for example, histograms of a particular feature like packet loss, jitter, etc. The
summarization method decreases the overhead and saves the bandwidth.

The paper is organized into three main sections. The first gives an overview of the currently used solution with appropriate mathematical descriptions. In particular, we focus on hierarchical aggregation. The following section covers a proposal of optimization of the hierarchical feedback aggregation in RTP/RTCP SSM. The optimization involves the mathematical background and then we consider some related constraining factors defined in the RTP/RTCP standard. The last section is devoted to the hierarchical tree establishment. This section covers a description of the feedback-level routing issue and its solution using a known binning algorithm. Then an overview of Tree Transmission Protocol follows with an example of message flow between session nodes.

II. RTP/RTCP PROTOCOLS

RTP protocol is intended to be used for real-time data transmission. It relies on UDP (User Datagram Protocol), see Figure 3, which offers connectionless and unreliable packet transmission transport service. The reason for using UDP is the protocol simplicity and hence a low level of delay in packet delivery. RTP itself does not assure QoS (Quality of Service). Usually, additional methods are used to assure the required QoS level, e.g. RVSP (Resource Reservation Protocol). The higher-level architectures such as H.323 and SIP utilize the services offered by RTP/RTCP.

![Frame header](IP) ![UDP](RTP) ![Data](Frame trailer)

Figure 3. RTP encapsulation

The RTP protocol is a signalling protocol that is primarily used for quality-of-service monitoring and congestion control. The goal is to keep signalling overhead as low as possible. Therefore it was specified in [1] that RTP can consume 5% of session bandwidth at maximum, in which transmission capacity is further divided into two parts, where the RR can consume 75% and the Sender Reports (SR) can consume the remaining 25% of the capacity allowed to the RTP. It means that all receivers share the bandwidth for receiver reports RR and it is clear that when the number of receivers is high, the period of RR becomes too long. 

When there is only 1 media source and \( n_T \) receivers in the SSM architecture, the service allocates bandwidth \( BW_T \), and the RR packet length is \( PL_{RR} \). The period of RR transmission can be expressed by the formula

\[
T_{RR} = \frac{PL_{RR} n_T}{0.0375 BW_T},
\]

(1)

The Sender Report is sent from the media source by the multicast, and the transmission period is given by the formula

\[
T_{SR} = \frac{PL_{SR}}{0.0125 BW_T},
\]

(2)

where \( PL_{SR} \) is the packet length of the sender report.

The standard RR packet size \( PL_{RR} \) can be estimated as \([1]: IP\ header – 20B, UDP\ header – 8B, RR\ header – 8B, RR\ report\ block\ (only\ one\ source) – 24B, SDES\ header – 4B, CNAME and padding – 28B. The result packet size is 92B (736b). When the Session Description data unit (SDES) is omitted then the resulting packet size is 60B = 480b.

To prevent very frequent RR transmission in the case of a small number of receivers a minimum transmission period of 5 seconds is recommended by the standard [1].

To judge the system behaviour for a large group of receivers let us assume the following example situation: Video streaming service with \( BW_T = 1\ Mbps, n_T = 10^7, PL_{RR} = 736b \). Formula (1) gives the result \( T_{RR} = 1963\) seconds, i.e. for more than half an hour. The actual reporting interval is randomly selected from the interval \(<0.5;1.5>^*T_{RR}, \text{ i.e. } <981;3044>\) seconds. This is too long for evaluation in the source because some pieces of information are too old to yield valuable results.

III. RTCP TRANSMISSION IN LARGE-SCALE MULTICAST - HIERARCHICAL ARCHITECTURE

To combat the problem a hierarchical structure for RTCP reporting was designed in [6]. The source and receivers are organized in the tree topology as shown in Figure 4.

![Source](Simple RR) \( i = 1 \)

Summarization node 

Summarization RSI 

End node

Figure 4. Tree topology of the RTCP feedback network

Receivers are divided into two groups: end nodes and summarization nodes. End nodes are the receivers, whose behaviour is the same as in the flat (one-level) architecture. Summarization nodes are new ones, whose function is to aggregate information from lower level nodes and to generate summarization reports that are sent to a summarization node at a higher level or to the source when the summarization node is at the highest level. In [7] a structure of a Receiver Summary Information (RSI) packet was specified and is shown in Figure 5.
Sub-report blocks contain distribution information about particular features like loss or jitter. The format proposed in [7] is shown in Figure 6. SRBT is the sub-report block type, the Length field expresses the length of the block in 32-bit words, NDB specifies the number of distribution blocks, and MF is a multiplicative factor that is used for the compression of the bucket size by a factor $m = 2^{MF}$.

The maximum number of distribution buckets is 256 for the packet loss and 512 for the jitter as specified in standard RTCP. The size of each bucket in bits depends on the number of receivers that are below the summarization point in the hierarchical structure, and on the multiplicative factor MF (see Figure 6). The overall length of the RSI packet can be calculated as follows: IP header – 20B, UDP header – 8B, RTCP/RSI header – 16B, packet loss SRB header - 12B, delay jitter SRB header - 12B, loss distribution – DL bytes, and jitter distribution – DJ bytes. The bucket size should be selected to carry the maximum number of nodes in the tree architecture below the summarization node divided by the multiplication factor $m$. When that number is $n_i$, then the size in bytes is

$$DL_i = 256 \left\{ \log_2 \left( \frac{n_i}{m_i} + 1 \right) \right\} = 32 \log_2 \left( \frac{n_i}{m_i} + 1 \right)$$ (3)

for the maximum number of buckets for the packet loss and

$$DJ_i = 512 \left\{ \log_2 \left( \frac{n_i}{m_i} + 1 \right) \right\} = 64 \log_2 \left( \frac{n_i}{m_i} + 1 \right)$$ (4)

for the maximum number of buckets for the delay jitter ($i$ specifies the level of the summarization node). The overall length of the RSI packet $PL_{RSI}$ in bytes is

$$PL_{RSI} = 68 + 96 \log_2 \left( \frac{n_i}{m_i} + 1 \right) \text{ (B).}$$ (5)

Provided the factor $m_i$ increases by a decreasing level $i$ in the tree, it is possible to keep the packet length almost at a constant value. To carry RSI information in one packet, whose maximum length given by the MTU parameter is typically 1500 B (in the Ethernet), it is necessary to meet the condition

$$BS = \log_2 \left( \frac{n_i}{m_i} + 1 \right) \leq 14,$$ (6)

where $BS$ is the bucket size in bits. Now when we calculate the size of a sub-report block we obtain

$$SRB_{\text{limit}} = 96*14 = 1344 \text{ (B).}$$ (7)

If we allow the number of buckets for packet loss and for delay jitter to differ from these maximum values, we can express the relationship between the maximum number of buckets and the maximum number of receivers that can be handled by a summarization packet with a length limit of 1500 B as shown in (8).

$$PL_{\text{limit}} = 68 + \frac{B_i + B_j}{8} \log_2 \left( \frac{n_i}{m_i} + 1 \right) \leq 1500 \text{ B.}$$ (8)

These relationships are shown in Figure 7. Variable ‘b’ is the bucket word length, $B_{\text{max}}$ is the overall number of buckets (both for packet loss and for jitter), and $N_{i,\text{max}}$ is the maximum number of receivers. It can be seen that, when for example 11-bit words are used in buckets, the maximum number of buckets is a bit more than one thousand, and the maximum number of addressed receivers is approximately 70 million. When the buckets are spread between packet loss and delay jitter, it is necessary to consider one constraint and this is that the maximum value in the length field (8 bits) of the sub-report-block header is 255 (Figure 6), which means that the maximum size of the sub-report block can be 255*4 = 1020 B (7). 12B of this is the header of the sub-report block.

The bucket size should be appropriate to the total bandwidth required by a service. When the maximum size $BS$ is selected, i.e. 14 bits, then the summarization report packet size according to the formula (5) will be 1412 B = 11296 bits. This large packet size decreases the number of summarization nodes in one group when the maximum report interval $T_{t,\text{max}}$ is set. As a result there will be more levels in the tree for a certain number of receivers and the delay will also be higher. This is even more restrictive for narrowband services where too low of a bit-rate is available for the RTCP.
Let us calculate the number of levels in the tree and the round-trip reporting interval. For simplicity, let us assume the tree structure is balanced, i.e. there are the same number of nodes at any level below each summarization node. Providing the summarization nodes are also the stream receivers, then level one has \( n_0 \) nodes, level two has \( n_0^2 \), level three has \( n_0^4 \), and so on. The total number of receivers \( n_I \) can be calculated as

\[
n_I = \sum_{i=1}^{I} n_0^i - 1 = \frac{n_0^{I+1} - 1}{n_0 - 1} \tag{9}
\]

We usually know the bandwidth for RTCP, the total number of receivers, and according to the packet length and the acceptable report packet rate we also select the size of the \( n_0 \), i.e. the number of receivers under the summarization node at the higher level (group number), and we would like to know how many levels the tree will contain and how large a round-trip delay \( T_{RT} \) (the delay between RR transmission by the bottom-most receiver and summarization report from the sender) will be reached. The number of levels \( I \) can be obtained from formula (9)

\[
I = \log_{n_0} \left[ \frac{(n_I + 1)(n_I - 1) + 1}{n_I - 1} \right] - 1 \tag{10}
\]

When we simplify the situation by assuming that the reporting interval is the same at all levels, the worst-case round-trip delay \( T_{RT} \) can be calculated as follows:

\[
T_{RT} = T_{RR} I + T_{RS} =

= K n_0 \log_{n_0} \left[ \frac{(n_I + 1)(n_I - 1) + 1}{n_I - 1} \right] + \frac{P_{RS}}{0.0375BW_R}, \tag{11}
\]

where

\[
K = \frac{P_{RR}}{0.0375BW_R} \tag{12}
\]

TRR is the average value of reporting interval, TRS is the summarization transmission interval of the sender and PLRS is the length of the summarization packet sent by the sender through the multicast back to the receivers.

Let us show the result of hierarchical structure using an example. Let the session bandwidth \( BW_R = 1 \text{ Mbps} \) and the number of receivers be \( n_I = 10^5 \), \( n_0 = 32 \), \( P_{RR} = 500 \text{ B} = 4000 \text{ b} \). From formula (10) we see that \( I = 3.33 \), i.e. a 4-level tree will be established. \( T_{RR} = K n_0 = 3.4 \text{ seconds} \). \( T_{RS} \) for one source is approximately half a second, so that the total round-trip delay is 14 seconds for the worst case. When compared with the flat (one level) architecture (1963 seconds) the result is approximately 140 times better. But the delay calculation is for the case when the 4-level tree is fully loaded. Actually the tree for the number \( n_I = 10^5 \) user will not be complete and level 1 will contain only 3 summarization nodes (full 4-level tree has capacity more than 1 million subscribers). Therefore the actual roundtrip delay will be even shorter, at only 11 seconds.

IV. HIERARCHY STRUCTURE OPTIMIZATION

In the previous example the variable \( n_0 \) was selected, but, as formula (11) shows, this parameter can be optimized. The goal is to find such \( n_0 \) for which formula (11) is minimal. When we differentiate this formula with respect to \( n_0 \), we obtain quite a complicated expression (13), which is a bit cumbersome for additional calculations.

\[
\ln \left[ \ln \left( n_I + 1 \right) \left( n_I + 1 \right)^{1/2} \right] + \frac{n_0 (n_I + 1)}{(n_I + 1)(n_I + 1) + 1} \ln n_0 - \ln n_0 - \ln n_0 = 0 \tag{13}
\]

To get a simple expression and to reach the result it is necessary to make some simplifications. First, let us simplify formula (9) to form

\[
n_I = n_0^I \tag{14}
\]

The error caused by that simplification is related to the inverse of \( n_0 \). The number of level I has a much simpler form

\[
I = \log_{n_0} n_I \tag{15}
\]

Formula (11) changes to the form

\[
T_{RT} = T_{RR} I + T_{RS} = K n_0 \log_{n_0} n_I + T_{RS} =

= K n_0 \ln n_I + T_{RS} \tag{16}
\]

After differentiating the previous expression we set the result equal to zero to find an extreme

\[
\frac{\partial T_{RT}}{\partial n_0} = K \ln n_I \frac{\ln n_I - 1}{\ln n_0} = 0 \tag{17}
\]

Equation (17) will be fulfilled when \( \ln n_0 = 1 \) and this is true for \( n_0 = e = 2.71828 \), i.e. when the number of members in a group is around that value, e.g. 2 or 3. The second derivative gives the expression

\[
\frac{\partial^2 T_{RT}}{\partial n_0^2} = K \ln n_I \frac{2 - \ln n_I}{n_0 \ln n_0} \tag{18}
\]

which is positive for \( n_I = e \), i.e. the extreme is the minimum.

When we choose \( n_0 = 3 \), formula (10) results in \( I = 10.74 \), i.e. 11 levels in the tree. \( T_R = Kn_0 = 0.32 \text{ seconds} \) and the total delay is approx. 3.6 seconds, i.e. 4 times shorter than what we calculated with \( n_0 = 32 \). For \( n_0 = 2 \) formula (10) results in \( I = 17.19 \) (18 levels) and the total delay is 3.9 seconds, so a bit more than for \( n_0 = 3 \).

The result obtained by the solution of equation (17) is interesting in that the minimum does not depend on the total number of receivers \( n_I \). This fact is also proven by simulation results obtained by Matlab and shown in Figure 8, which shows the dependence of the total delay on the total number of receivers \( n_I \) and on the number of receivers in one group \( n_0 \) placed in the hierarchical structure. For the calculation, the exact formula (11) was used.

![Figure 8. Overall delay in the tree versus total number of receivers \( n_I \) and the number of receivers in one group \( n_0 \)](image)
A. Considerations of some constraints

Previous calculations are an ideal example of where the node processing times and network transmission delay are considered zero. Also, these examples do not consider some limitations given by the standard [1] like a minimum reporting interval that was set to 5 seconds. This means that the group size will be specified by this interval limit, the maximum length of the reporting interval, and the RR or RSI packet size.

Let us consider three constraints:

- 5 second limit,
- different lengths of receiver reports and summarization packets,
- summarization nodes are special nodes for the summarization of receiver reports and not receivers of the media stream.

Provided that the size of the summarization packets does not change within the whole tree, the balanced tree will contain the same number of summarization nodes below the higher level summarization node at all levels. In this case the equation (9) changes into the form given by formula (19)

$$n_i = n_i^{(i-1)} n_b,$$

where $n_b$ is the number of summarization nodes below the next higher level summarization node.

The calculations of the number of tree levels also changes to form

$$I = \log_n(n_t/n_b) = 1 + \frac{\ln n_t - \ln n_b}{\ln n_b}.$$  \hspace{1cm} (20)

Then the round-trip delay can be calculated as

$$T_{RT} = T_{RR} \geq 5s \geq (I-1)T_{RS} \geq 5s + T_{RS} =$$

$$= \frac{1}{BW_{RR}} \left[ n_b P_{RR} + n_b P_{RLE} \ln n_t - \ln n_b + P_{RS} \right]$$

where $BW_{RR}$ is the bandwidth for receiver report transmission. When we consider the same group delays at the bottom level (receiver level) and also at all other levels (summarization levels), it will hold that

$$T_{RR} = T_{RS},$$

$$n_b P_{RR} = n_b P_{RLE},$$

$$n_b = \frac{P_{RR}}{P_{RLE}}.$$  \hspace{1cm} (22)

Then the formula (21) changes to

$$T_{RT} = \frac{1}{BW_{RR}} \left[ n_b P_{RR} \left(1 + \frac{\ln n_t - \ln n_b}{\ln n_b} \right) + P_{RS} \right] =$$

$$= \frac{1}{BW_{RR}} \left[ n_b P_{RR} \left(1 + \frac{\ln n_t - \ln n_b}{\ln n_b + \ln P_{RLE} + \ln P_{RLE}} \right) + P_{RS} \right].$$  \hspace{1cm} (23)

Provided the actual receiver position in the Internet is not considered (this problem is solved in chapter V) the balanced tree establishment will be realized as shown in Figure 9 a), b), c), d), e):

When the total number of receivers $n_t$ is smaller than or equal to $n_{RS}$ (this is the number of receivers when the $T_{RR}$ is equal to 5 seconds) the worst-case delay from the receivers to the source is fixed and equal to 5 seconds (Figure 9 a)). When $n_t$ exceeds $n_{RS}$, but is less than $2*n_{RS}$, the delay is within the interval (5, 10) seconds (Figure 9 b)). When $n_t$ crosses number $2*n_{RS}$, the first set of summarization nodes appears, at the beginning and exactly 3 nodes will be added, 2 of them will have $n_{RS}$ receivers and the third one will have the rest of them. The delay will be again constant – 10 seconds. The situation is shown in Figure 9 c). Variable $n_{RS}$ is the number of summarization nodes in one group for which the period of summarization reports reaches 5 seconds. Further increase of the number of receivers to $n_t = 2*n_{RS}*n_{RS}$ and the tree keeps only one level of summarization nodes (see Figure 9 d)) and the delay is within the interval (10, 15) seconds. When $n_t$ exceeds the value $2*n_{RS}*n_{RS}$ the worst-case delay is again constant and equal to 15 seconds (Figure 9 e)). In the same way the other levels in the summarization tree are added when $n_t$ increases further.

Figure 9. Steps in tree establishment
The relations between the total number of receivers $n_T$ and the delay for different service bitrates can be seen in Figure 10. In addition to the bitrates other initial conditions were set:

- PLRR = 736 bits,
- number of jitter buckets $B_j$ was set to 512, and number of packet loss buckets $B_l$ was set to 256 and
- number of bits for each bucket was set to 8.

![Figure 10. Receiver Report Transport delay versus total number of receivers](image)

The length of the summarization report given by formula (5) is 6688 bits. Let us chose the case when $BW_T$ is $10^{10}$ bps. Then the parameter $n_{Lat}$ is 255 and $n_{SS}$ is 29. When we are interested in the delay for $n_T = 10^5$ we can see from Figure 10 that the resulting delay is 15 seconds and the number of levels in the tree is 3.

V. ESTABLISHMENT AND MANAGEMENT OF FEEDBACK TREE

A. Feedback tree Establishment

In this section, we introduce a feedback tree establishment algorithm for the RTCP feedback transmission within a multicast session. The feedback tree transmission structure should be formed in a way that keeps the round-trip delay as low as possible. However, using a previously introduced number of group members only to arrange the tree structure is not satisfying in terms of data transmission in IP networks. In order to achieve a good routing performance, session members should be also organized in groups considering the relative distance between them and their feedback target. This relative distance should be kept as small as possible. In large-scale multimedia distributions, an uncontrolled partitioning of members into groups could lead to an inefficient IP-level routing. For example, it could happen that a member located in one country reports its feedback to a summarization node in another country, whereas the next higher summarization node is situated again in the original country. This leads to a difference in the feedback-level (or application-level) overlaying topology and the IP-level underlying topology.

The intuitive solution of the feedback-level routing problem is to integrate the exact IP-level routing information into the tree establishment algorithm. This solution is however quite tricky since a close cooperation with all the routing protocols involved is required. Instead, we thought of another known solution that uses the network latency feature to find the appropriate closest nodes on the Internet [1]. The algorithm called “binning” partitions the nodes into bins, and the nodes within a bin are thought to be relatively close in terms of network topology. Results presented in [1] and [14] show that only approximate tree structure information offered by the binning algorithm allows a significant improvement of the routing complexity. The algorithm is based on a set of landmark nodes (LM) placed on the Internet in a certain way. Then, a node evaluates its round-trip time (RTT) to these landmarks in a specified period of time. On the basis of the RTTs measured, nodes join bins as follows: each node creates a vector consisting of landmarks ordered by increasing RTTs. The resulting landmark vector then defines a bin, and nodes with the same vector belong to the same bin. Furthermore, nodes can be partitioned into bins on the vector similarity basis. An important feature of this algorithm is that nodes assign themselves into bins without any communication with the other nodes. This is ideal for large-scale multimedia sessions, where communication among all nodes would be harmful to the network bandwidth load, provided that thousands or millions of nodes are involved. As an algorithm example, see Figure 11 for landmark vector creation.

![Figure 11. Landmark vectors](image)

In order to put the feedback transmission tree structure into effect, every session node should know a feedback target (FT)/summarization node on the tree level above to which to send its feedback. To assure this, the Feedback Transmission Manager (FTM) is used. The FTM builds the tree structure based on specific rules and information available from all session nodes. The FTM keeps the tree structure up-to-date by rebuilding the tree when new information is available. When the tree structure is established, it communicates with all session nodes to inform them to which feedback target they should send their feedback. As mentioned previously, two rules for tree establishment are used: 1) number of group members below a feedback target and 2) distance between nodes and their feedback target.

The FTM assumes general availability of a set of dedicated servers acting as feedback targets. These dedicated servers should be typically controlled by a service provider offering media broadcasting services such as IPTV. An example scenario of shared feedback targets is shown in Figure 12. We suppose that a provider manages several IPTV streams from different TV stations.
Each media stream involves a main feedback target (MFT), which is the top feedback target in the tree hierarchy. The MFT and the actual media source could be in fact one piece of hardware. A feedback transmission tree is organized below each main feedback target. The middle layer consists of a set of dedicated servers working as feedback targets. These feedback targets are shared among several feedback trees. It depends on the feedback target performance how many trees it can be involved in. However, feedback targets higher in the feedback tree hierarchy are expected to be more loaded with feedback processing. Finally, the lowest layer consists of end nodes acting as receivers of a particular IPTV stream. One end node can receive several media streams and consequently belong to several feedback trees.

B. Tree transmission protocol

For the purpose of tree structure information transmission between the FTM and session nodes (feedback targets and end nodes), the Tree Transmission Protocol (TTP) has been proposed. The TTP protocol overview and its position in protocol stack are depicted in Figure 13. The key idea used in the TTP is as follows: Let us suppose that the FTM is aware of the landmark RTT vectors of all nodes, including both feedback targets and end nodes. The vector is formed as a list of landmarks starting with the landmark with the lowest RTT value measured. To assure that the FTM knows all vectors, each node sends its vector to the FTM periodically. Then the FTM is able to establish the tree structure in terms of relative distance between feedback targets and its group members. The FTM also meets the second requirement (number of group members) by selecting suitable feedback targets from a set of available dedicated servers. The ratio of the significance of relative distance to the number of group members can be set on FTM. In this way, the required tree structure is locally established at the FTM. If the FTM cannot achieve the required tree structure by selecting feedback targets from the available set, it could also use fake landmark vectors of feedback targets.

Figure 14 shows three main entities within the TTP protocol and their mutual relationships. The FTM sends the tree organization information to all nodes using the existing SSM channel. Furthermore to work properly, the FTM must be aware of the LM RTT vectors of all nodes. To assure this, all nodes (including feedback targets) send this information via unicast connections to the FTM. Finally using unicast connections, the FTM communicates with all involved feedback targets in order to set their participation in a specific tree (FT activation channel).

A description of TTP protocol messages is shown in Tab. 1. All the messages are based on a general message structure consisting of an important field called Feedback Tree Identifier (FTID). The purpose of FTID is to differentiate feedback trees within a provider space. The FTID value could be, for example, the Synchronization Source Identifier (SSRC) of the originator of the packet within a RTP/RTCP session. Using the SSRC value, different tree structures can be established for each synchronization source.

<table>
<thead>
<tr>
<th>TABLE I. OVERVIEW OF TTP MESSAGES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feedback Target Activation Packet (FTA) - information about requested resources (number of nodes which are expected to send feedback to a feedback target, priority of resource reservation, average data size of packets carrying the feedback data, ...) - specifies a session information server for allowing feedback targets to obtain the SSM channel description (S,G), for example using SDP protocol</td>
</tr>
<tr>
<td>Feedback Target Confirmation Packet (FTC) - informs the FTM about accepting/denying its participation in a feedback tree</td>
</tr>
<tr>
<td>Landmark Set Packet (LMS) - information for binding landmark ID and IP address/port - list of landmark IP addresses/ports for session nodes to evaluate the LM RTT vector</td>
</tr>
<tr>
<td>Feedback Target Specification Packet (FTS) - specifies feedback target for a group of end nodes - information about all groups in the session (group size, LM RTT vector specification, LM RTT vectors, ...)</td>
</tr>
<tr>
<td>Landmark Vector Packet (LMV) - carries LM RTT vector from session nodes to the FTM</td>
</tr>
</tbody>
</table>

Figure 14. TTP main modules

The state diagram of an end node working with the TTP protocol is shown in Figure 15. The two timers used serve as triggers for the evaluation and transmission of the LM RTT vector. This process involves (timer #1) sending ping requests to landmark nodes defined in the available landmark set. The second event (timer #2) is used to balance, in time, the traffic load. The timer #2 expiration interval is a function of the number of end nodes within a session. With a greater amount of end nodes, the timer
expiration time is longer. In this way, we are able to assure that the MFT, which the LMV packets are sent to, is not overloaded with these packets from all session nodes at the same time. On the other hand, the LMV packets are sent to the MFT as quickly as possible to obtain a better feedback tree convergence time. The third event in the state diagram describes receiving the LMS packet, which carries a set of landmarks used for LM RTT vector evaluation. The next state is described by receiving the FTS packet which carries information about a group this node participates in. This information also includes the specification of a feedback target where the feedback from this group should be sent to (IP address/port). To extend this state diagram for a feedback target, another state is added (shown with dashed lines). This state involves receiving the FTA packet for the feedback target activation and consequently sending the FTC packet carrying the activation result.

The transmission scenario shown in Figure 16 shows the basic communication between the FTM with a feedback target and an end node. The corresponding feedback tree structure therefore consists only of these tree entities. For simplicity, we omitted the communication involving the distribution of session information to nodes interested in media reception. A multimedia session is established when a media source registers an SSM multicast session defined by the source address (S) and the multicast group address (G). For that purpose the Session Description Protocol (SDP) [2] can be used. Also for simplicity, the messages covering nodes joining the multicast group are omitted.

![Figure 15. Basic TTP state diagram](image)

After both the media source and the FTM have joined the multicast group (S, G), the FTM attempts to activate the required feedback target for the feedback transmission. For that purpose, the FTA packet is used. If the feedback target accepts the activation (carried in an FTC packet), it sends an SDP request to obtain a multicast session description (S, G). The feedback target then also receives an LMS packet with a set of landmarks to be used to form an LM RTT vector. Consecutively, an LMV packet including the LM RTT vector is sent to the FTM. When an FTS packet is received by the FTM, the feedback target is involved in the feedback tree. In order to stop the feedback target participation in the feedback tree, an FTA packet is sent with the appropriate flag set.

When an end node becomes a member of the session, it starts to report its feedback on media reception directly to the main feedback target specified in the session description previously disturbed by the SDP. In this way no immediate communication with the FTM is required when end nodes join a session. This feature is quite important in a case involving massive session joining, for example, when an interesting IPTV program is beginning to start. After a specific time, which depends on the LMS packet transmission period, the end node receives information about an available set of landmarks. Using this information, the end node evaluates its own LM RTT vector and sends it to the FTM in an LMV packet. After another time interval, the end node receives the FTS packet with information identifying a new feedback target to be used. Then the communication process continues by sending periodically the LMS, FTS and LMV packets. The periodic LMV packet transmission allows the FTM to check whether the end node is still participating in the session (shown as LMV timeout).

**VI. RTCP TRANSMISSION IN MOBILE NETWORKS**

It is likely that streaming services will play an important role also in mobile networks. When we want to implement the tree structure architecture it is necessary to change the strategy. It would not be possible to use any mobile equipment to do the function of the summarization node. The idea is that this function should be done by the SGSN (Serving GPRS Support Node) nodes. The GGSNs (Gateway GPRS Support Node) could be the second-level summarization nodes. The situation is shown in Figure 17. The disadvantage of this solution is that it is necessary to modify the GSN nodes (to add some software) and that the GSN nodes have to do the function of summarization node for a number of different sessions simultaneously. On the other hand, the tree structure within the mobile network is fixed and therefore easy to establish.

Of course there are also some problematic factors in the mobile networks: limited capabilities (low processing power, low input buffer), hostile wireless communication environment where conditions change continuously (radio link throughput), and also a handover. These problems
should also be considered in RTCP protocol modifications for mobile network environments. RTCP messaging should be included in the Session management part of control functions implemented in the packet oriented part of the network.

Figure 17: Tree structure implementation in a mobile network environment

VII. EXAMPLES OF USING THE FEEDBACK TREE TRANSMISSION

The feedback tree transmission can be used in the following areas:

- **Service Level Agreement** - The feedback on a receivers' reception quality can be used by a provider/subscriber to check whether their service contract is fulfilled. The gathered feedback information can also be easily made public on-line for potential new subscribers to see if the service offered meet their needs.

- **Polling** - Information sent by receivers can also involve any non-reception-quality related data. An example is gathering the statistical polling data. In this way, users vote for several options offered and the results are available in a short time for further use.

- **Receiver localization** - Using a set of properly distributed landmarks, the system is also able to localize receivers with a given accuracy. An example is the location-based targeted advertisement to IPTV subscribers or location-related information for poll results.

- **Quality of Service** - The feedback on receivers' reception quality offers valuable information for QoS improvements. For example, receivers with a temporary poor quality of reception can be automatically redirected to a media stream that better suits their current network capabilities.

VIII. CONCLUSION

The paper dealt with the problem of large-scale RTCP feedback in multimedia sessions. A hierarchical structure was proposed to solve the problem of very long RTCP periods in large sessions with many subscribers. The round-trip delay in the tree feedback structure was mathematically specified. Examples were presented and compared with standard flat architecture. The results show that this structure works well even in the case of million sessions. Optimum value for tree architecture and for a particular total number of receivers was calculated. It was found that this value did not depend on the total number of receivers. However, the optimal number of members in a group is too small (3 nodes in one group) and the optimization algorithm used does not consider the lowest allowed reporting interval when specified by the standard. This constraint was also considered in the other part of the article and delay calculations were done. Even in the case of a 5 second minimum RTCP period the overall delay was several tens of seconds for quite large sessions.

The problem of the RTCP feedback tree establishment was also solved. The method for finding the nearest summarization node in the IP network structure was designed. To manage the tree the new protocol was designed and specified. Now we design software modules representing RTCP end-nodes, summarization nodes and the source node to evaluate the proposed solution in the real network.

ACKNOWLEDGMENT

This paper has been worked out with the support of grant projects GACR 102/06/1569, AVCR 1ET301710508, AVCR 1ET301710510, and FRVS 850/2007/F1a.

REFERENCES


Vít Novotný, (born in 1969 in the Czech Republic)

He graduated from the Department of Telecommunications of the Faculty of Electrical Engineering and Computer Science at the Brno University of Technology in 1992, the Ph.D. degree received the Brno University of Technology in 2001 in “electronics and communication technologies”, the assistant professor degree received at the Brno University of Technology in 2005 also in the area of “electronics and communication technologies”.

In the past he did the research in the areas of non-filtering applications of switched capacitors and of the current and voltage conveyors. Current professional interests are mobile and integrated packet data networks, their services and terminal equipment.

Now he works with the Dept. of Telecommunications, Brno University of Technology, Purkynova 118, 612 00 Brno, Czech Republic where he entered in 1992 as an assistant.

Dan Komosný (born in 1976 in the Czech Republic)

He graduated from Brno University of Technology, Faculty of Electrical Engineering and Computer Science in the field of Electronics and Communications (2000), Ph.D. in Teleinformatics (2003).

He is engaged in research focused on voice transmission over IP networks (VoIP). He also focuses on the development of e-learning tools using formal and visual languages – the SDL object-oriented design language and the MSC trace language. Now he works with Dept. of Telecommunications, Brno University of Technology, Purkynova 118, 612 00 Brno, Czech Republic.