|  |  |  |
| --- | --- | --- |
| 3GPP TSG System Aspects WG4#38  Rennes, France 13-17 February 2006 | | S4-060157 |
| Source: | SQ Subworking Group**[[1]](#footnote-2)** | |
| Title: | Background on Error Conditions for the Characterization in TR 26.936 | |
| Document for: | Information | |
| Agenda Item: | 12 | |

# Introduction

This document explains the tools and methods which have been employed to simulate the 3GPP transport conditions for MBMS streaming delivery. The application layer FEC as specified in TS 26.346 was not used in this characterization.

# Simulation Approach

## Overview

The processing chain which was used for these tests aimed at resembling a real-world transmission scenario. This includes source encoding, RTP packetization, mapping to RLC-PDUs, de-packetization and decoding. Error insertion was performed on the RLC-PDU level using a file-based approach and simulator software for GERAN and UTRAN which is briefly described in the following and is also included in the zip-archive in Annex A. In principle, all RTP packets which were affected at least partly by a RLC-PDU error were discarded. The simulation also modeled the application of Header Compression by reducing the RTP/UDP/IP packet headers to some constant number of bytes. All packet loss rates in TR26.936 mentioned refer to RLC-PDU packet loss rates, not to RTP packet loss rates. Some indication on the resulting RTP loss rates are provided in section 3.

The attached SA4-internal tool simulation tool sa4sim (see Annex A) was used to map RLC-PDU error traces to RTP packet losses. A block diagram of this software is shown in Figure 1.



Figure 1 RTP Packet Loss Simulator. Link level behaviour is captured in error traces and mapped to RTP packets. The input format to the simulator is according to the interim file format (IFF).

The audio codecs produce RTP packets which are encapsulated into an interim file format (IFF). The resulting packet stream is stored in the IFF and can be processed by the simulator. The IFF is such that each RTP packet is preceded by two additional fields as shown in Figure 2. The fields include the size of the RTP packet including payload and header in bytes and a time stamp (time in ms). This time stamp is used for different purposes: It basically indicates when the included RTP packet is virtually available for the next processing unit. This file format allows the encoder to signal to the simulation software when the packet is available and also the simulation software when the packet is available to the decoder.

RTP Payload

packet size

RTP Header

time stamp

Figure 2. Audio codec interface to the packet loss simulator. In addition to the RTP packet, two additional header fields are generated by the codecs. Packet size: size of the RTP packet including payload and header. Time stamp: time (in ms) at which the packet is transmitted or received.

In addition to the modification of the time stamp, the software tool maps the RTP packets to the error masks on RLC-PDU layer. One PDU error results in that all RTP packets which are partially or completely mapped to the lost PDU, to be discarded. Discarded packets are not further signaled to the decoder, but are just not written in to the output file. All non-discarded packets are error-free. It was expected that the audio decoder including the de-packetizer is able to detect packet losses by only employing the information being included in correctly received RTP packets.

## Software Configuration

The software tool can be configured by different parameter settings. An exemplary configuration file sa4sim.cfg is provided. In the following we will briefly define the input parameters:

RTPinfile = "infile.iff" # Input File

RTPoutfile = "outfile.iff" # Output File

LogFile = "log.txt" # Log File

Bearer = 1 # Bearer Number

RandomSeed = 0 # Random Seed

ErrorFreeRTP = 0 # Number of error-free RTPs

TSModeSender = 2 # 1 ignore TS at transmitter,

# 0 use TS, send dummy,

# 2 same 0, but non-alignment

StatFile = "stat.dat" # Statistics File

The software can be executed for example by using

sa4sim –f sa4sim.cfg –p RTPinfile=<user defined> -p Bearer=<1-10> -p RandomSeed=<1-32>

The configuration parameters are explained in more details in the following.

**RTPinfile**:

File name for the input file with interim file format. The file should consist of a packetized format as follows:

* 4 bytes length indication of following payload *L*1 in bytes (32-bit unsigned integer),
* 4 bytes timing information (generation time in ms, 32-bit integer),
* *L*1 bytes payload being an RTP packet including the RTP header,
* 4 bytes length indication with *L*2,
* 4 bytes timing information,
* *L*2 bytes payload being an RTP packet including the RTP header
* etc.

The timing information indicates the time instant in ms when the packet is released by the encoder. The simulator can use this information to maintain a certain sending timeline and to send dummy data or data from other applications in case no data is available from the considered audio stream or to drop audio stream packets in case of buffer overflow at the transmitter. This allows for example the simulation of live encoding.. The use of this information is optional and is indicated by TSSenderMode (see below).

**RTPoutfile:**

File name for output file also in interim file format. It is identical to the input file except that

* Entire RTP packets including length information, timing information, and payload might be missing in case that the RTP packet has at least partly been mapped to a lost RLC-PDU.
* The timing is altered such that the time instant in ms is provided when the RTP packet is released by the receiver.

**Log File**:

Logs events in the simulator.

**Bearer**:

A specific bearer can be selected using a number which addresses a bearer. The bearer is further specified in a file named <bearers.txt>. In this file each non-commented line (comment is #) represents a bearer. Additional bearers can be added. More details:

# This file contains some bearer configuration. The bearers can be indexed by the number.

# The specific columns are explained in the following

# Number: Number of the bearer used as index (integer)

# File: File name of the error masks, can be bit errors or packet errors

# Format: Gives the format of the file (binary for bit errors, ascii for packet errors)

# TTI: Transmission Time Interval in ms

# RFS: Radio Frame Size in bytes describes the RLC-PDU size

# Mode: Transmission Mode: 1 is acknowledged bearer, 0 is unacknowledged bearer

# System: 0=CDMA2000, 1=UMTS, only difference is in sizes of fields added for headers

# CRUIH: Compressed RTP/IP/UDP header size assuming header compression

# RDel: (only for ACK mode) The retransmission delay before it is available at the

encoder in multiples of the TTI

# Amod: (only for ACK mode) Mode of Acknowledged Bearer (0=persistent, 1=non-persistent)

# NoRet: (only non-persistent ACK mode) Number of Retransmission for ACK mode

#

# The following bearers are defined

# Number File Format TTI RFS Mode System CRUIH

# MBMS EGPRS Bearers

1 EGPRS/EP0.txt ascii 15 74 0 3 8

2 EGPRS/EP1.txt ascii 15 74 0 3 8

3 EGPRS/EP2.txt ascii 15 74 0 3 8

4 EGPRS/EP3.txt ascii 15 74 0 3 8

5 EGPRS/EP4.txt ascii 15 74 0 3 8

6 EGPRS/EP5.txt ascii 15 74 0 3 8

7 EGPRS/EP6.txt ascii 15 74 0 3 8

8 EGPRS/EP7.txt ascii 15 74 0 3 8

# MBMS UTRAN Bearers

11 MBMS\_32kbps\_80ms\_VA3\_BLER\_1\_0.txt ascii 80 320 0 1 4

12 MBMS\_32kbps\_80ms\_VA3\_BLER\_5\_0.txt ascii 80 320 0 1 4

13 MBMS\_64kbps\_80ms\_VA3\_BLER\_1\_0.txt ascii 80 640 0 1 4

14 MBMS\_64kbps\_80ms\_VA3\_BLER\_5\_0.txt ascii 80 640 0 1 4

**RandomSeed**:

Integer value to modify the starting position in the error pattern. For longer simulations it is proposed to start with 0 and increment the value by 1 for each run. Furthermore, this value is used as the initial seed of the random generator.

**ErrorfreeRTP**:

Specifies a certain number of RTP Packets at the beginning of the file which can be forwarded directly to the receiver without being lost. This is especially important if for example the first packet contains setup information, but was not used in the audio characterization tests.

**TSModeSender**:

* If 0, the timing information in the packets is ignored.
* If 1, the program evaluates the timestamp and does not send the packet until the internal clock is as least as high as the time stamp of the packet. Therefore, one can simulate live encoding.
* If 2, just as 1. In addition, with the start of a new RLC-PDU, the start of the RTP packet is not aligned with the start of the RLC-PDU, but a random offset (uniformly distributed with the RLC-PDU length) is chosen. This mode was used in the audio characterization tests.

**Stat File:**

Provides some information for each run. In detail, the following information is provided:

* Date and time
* Bearer number
* Starting position in error file determined by RandomSeed
* Bit error rate for this transmission
* Native RLC-PDU loss rate
* Effective bit rate counting only correctly received RLC blocks
* Total Number of RLC frames
* Total Number of retransmitted RLC frames
* Total Number of dummy RLC frames
* Total Number of RTP packets excluding the first error free RTP packets
* Total RTP packet loss rate
* The bit rate for the video in kbit/s
* The total amount of time to transmit this file in ms.

## Error Traces for GERAN

RLC-PDU error traces masks for GERAN were generated using the following parameters:

* Transmission mode: Non-persistent mode with blind re-transmission without feedback.
* Receiver mode: Incremental Redundancy (IR)
* Frequency hopping scheme: Ideal
* Statistical independence of RLC block errors.
* Modulation and coding scheme: MCS-6
* Number of RLC block re-transmissions: 2 (in total 3 transmissions)
* Bitrate for using 4 multislots: 39.47 kbit/s

## Error Traces for UTRAN

RLC-PDU error traces masks for UTRAN were generated using the following parameters

* RLC-PDU loss rates: 1% & 5%
* Bearer bitrates: 32 & 64 kbps
* Geometry: -3dB
* Selective combining with two radio links
* Slot formats: 8 for 32 kbps and 10 for 64 kbps (see S4-040803 for details)
* Channel model: Vehicular-A, 3 km/hr
* TTI: 80 ms

It was assumed that the PDU sizes were 320 and 640 bytes for 32 and 64 kbps bearers, respectively. For more details on the generation of the error traces it is referred to document S4-040803. The above simulation parameters have been agreed with RAN4.

# Simulation Parameters and Results

For the simulations bearers 3, 7, and 8 are used for EGPRS and 11, 12, 13, and 14 are used for UTRAN. Table 1 shows the average RTP loss rate for files of length 10 seconds with RTP packets every 20ms with constant RTP packet size. The average for RandomSeed set to 0,1,2,…31 is evaluated. Note that the RTP loss rate is in general slightly higher than the target RLC-PDU rate due to segmentation losses. The Unix batch file rtp\_loss\_rate as well as some artificial files with appropriate bitrates in iff-format are included. To verify the results of the columns please run the rtp\_loss\_rate <bearer> <bitrate in kbit/s>.

Table 1 Average RTP loss rate for different bearers and media bitrates.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Bearer →  Bitrate ↓ | EGPRS 1%  EP2: 3 | EGPRS 6%  EP6: 7 | EGPRS 10%  EP7: 8 | UTRAN 1%  11/13 | UTRAN 5%  12/14 |
| 16 kbit/s | 1.29% | 8.81% | 14.07% | - | - |
| 20 kbit/s | 1.48% | 9.95% | 15.71% | 0.95% | 5.76% |
| 24 kbit/s | 1.71% | 10.85% | 17.32% | - | - |
| 32 kbit/s | - | - | - | 1.18% | 6.36% |
| 40 kbit/s | - | - | - | 0.89% | 6.14% |

# Annex A: Software Archive sa4sim.zip

Included in zip-archive.

1. Contact: Thomas Stockhammer ([stockhammer@nomor.de](mailto:stockhammer@nomor.de)). BenQ mobile. Munich. [↑](#footnote-ref-2)