Presentation of Specification to TSG SA

Presentation to: **TSG SA Meeting #21**

Document for presentation: **TR 26.937 RTP Usage Model, Version 2.0.0**

Presented for: **Approval**

Abstract of document:

The characterisation activity consists mainly of showing the expected PSS Release 5 performance in different use cases and network conditions and is expected to reveal any weaknesses and/or optimisation possibilities. The PSS characterisation results should serve as problem definition and requirements, based on which algorithmic enhancements can be defined for possible inclusion in PSS Release 6.

Changes since last presentation to TSG SA Meeting #18:

The TR has been refined and completed.

Outstanding Issues:

None.

Contentious Issues:

None.

3GPP TR 26.937 V2.0.0 (2003-09)

Technical Report

3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Transparent end-to-end packet switched streaming service (PSS); RTP usage model (Release 5)

Keywords
UMTS, IP, packet mode, protocol

3GPP

Postal address

3GPP support office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet
http://www.3gpp.org

Contents

Foreword

This Technical Report has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 1 presented to TSG for information;
  - 2 presented to TSG for approval;
  - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The objective of this document is to characterise the 3GPP Packet-switched Streaming Service (PSS). In doing so, the document considers the impacts of the underlying network configurations and how the streaming mechanism itself could be optimised.

The scope of this document includes consideration of (non-exhaustive):

- Trade-off between radio usage efficiency and streaming QoS
- Feedback of network conditions and adaptation of stream and/or the transmission of the stream
- Optimal packetisation of the media stream in line with the segmentation within the transport mechanism
- Error robustness mechanisms (such as retransmission)

Client buffering to ease the QoS requirements on the network and enable more flexibility in how the network transport resources are applied.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document.*

[1] 3GPP TR 41.001: "GSM Release specifications".

[2] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[3] 3GPP TS 26.234 (V5.0.0 onwards): "Transparent end-to-end packet switched streaming service (PSS); Protocols and codecs".

[4] 3GPP TS 23.107: "QoS Concept and Architecture".

[5] IETF RFC 3550: "RTP: A Transport Protocol for Real-Time Applications", Schulzrinne H. et al., July 2003.

[6] 3GPP TS 22.233: “Transparent end-to-end packet-switched streaming service. Service aspects (Stage 1)” (Release 5)

[7] 3GPP TS 25.322: “RLC protocol specification” (Release 5).

[8] V. Varsa, M. Karczewicz, Long Window Rate Control for Video Streaming, Proceedings of the 11th International Packet Video Workshop, 30 April – 1 May, 2001, Kyungju, South Korea.

[9] 3GPP TS 34.108: “Common test environments for user equipment (UE). Conformance testing” (Release ‘99).

[10] 3GPP TS 34.108: “Common test environments for user equipment (UE). Conformance testing” (Release 4).

[11] 3GPP TS 25.323: “Packet data convergence protocol (PDCP) specification” (Release 5).

[12] IETF RFC 3095: "Robust Header Compression (ROHC): framework and four profiles: RTP, UDP, ESP and uncompressed", C. Bormann (Ed.), July 2001.

[13] 3GPP TS 44.064: "Mobile Station – Serving GPRS Support Node (MS-SGSN); Logical Link Control (LLC) layer specification" (Release 5).

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3G TR 21.905 [2] and the following apply:

network: in the context of the RTP usage model network refers to the UMTS bearer service between the entry-point of the UMTS network (i.e. GGSN) and the UE.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [2] and the following apply:

2G	Second generation
AM	Acknowledged Mode
AMC	Adaptive Modulation and Coding
AMR	Adaptive Multi-Rate codec
AMR-WB	AMR WideBand
ARQ	Automatic Repeat ignall
BLER	Block Error Rate
CBR	Constant Bit Rate
CBRP	CBR Packet transmission
CN	Core Network
CS	Circuit Switched
DCH	Dedicated Channel
DL	DownLink
DSCH	Dedicated Shared Channel
DSP	Digital Signal Processing
EDGE	Enhanced Data rates for GSM Evolution
EGPRS	Enhanced GPRS
GERAN	GSM/EDGE RAN
GOB	Group Of Blocks
GPRS	General Packet Radio Service
GSM	Global System for Mobile communications
GTP	GPRS Tunneling Protocol
H-ARQ	Hybrid ARQ
HRD	Hypothetical Reference Decoder
HSDPA	High Speed Downlink Packet Access
HTTP	Hypertext Transfer Transport Protocol
IP	Internet Protocol
IR	Incremental Redundancy
ISDN	Integrated Services Digital Network
L2	Layer 2
LAN	Local Area Network
LLC	Logical Link Control
LWRC	Long Window Rate Control
MCS	Modulation and Coding Scheme
MTU	Maximum Transmission Unit
PCU	Packet Control Unit
PDCP	Packet Data Convergence Protocol
PDP	Packet Data Protocol
PDTCH	Packet Data Traffic Channel

PDU	Protocol Data Unit
PS	Packet Switched
PSNR	Peak Signal to Noise Ratio
PSS	Packet-Switched streaming Service
QCIF	Quarter Common Interchange Format
QoS	Quality of Service
QP	Quantization Parameter
RAB	Radio Access Bearer
RAN	Radio Access Network
RLC	Radio Link Control
RNC	Radio Network Controller
ROHC	Robust Header Compression
RRM	Radio Resource Management
RTCP	RTP Control Protocol
RTP	Real-time Transport Protocol
SDP	Session Description Protocol
SDU	Service Data Unit
SMIL	Synchronized Multimedia Integration Language
SNDCP	Subnetwork Dependent Convergence Protocol
SW	SoftWare
TCP	Transmission Control Protocol
TFRC	TCP Friendly Rate Control
TMN	Test Model Near-term
TTI	Transmission Time Interval
UDP	User Datagram Protocol
UE	User Equipment
UL	UpLink
UTRAN	UMTS Terrestrial RAN
VBR	Variable Bit Rate
VBV	Video Buffering Verifier
VBRP	VBR Packet transmission

4 Background and motivation

The characterisation activity consists mainly of showing the expected PSS Release 5 performance in different use cases and network conditions and is expected to reveal any weaknesses and/or optimisation possibilities. The PSS characterisation results should serve as problem definition and requirements, based on which algorithmic enhancements can be defined for possible inclusion in PSS Release 6.

5 Overview

Void.

6 End-to-end PSS system

When considering use cases for 3GPP PSS, an end-to-end system and protocol view is taken into consideration. For instance, the following issues are taken into account:

1. Multimedia content creation;
2. Streaming server media transmission and traffic characteristics;
3. UMTS QoS profile parameters and their implications;
4. Bearer and Layer 2 network protocol options (including PDCP and RLC);
5. Network transport channel mapping (dedicated or shared channels);

6. Core network;
7. Streaming client.

The PSS use cases assume the streaming server to be located in the mobile operator's network or connected to the mobile network over the Gi interface where sufficient QoS is available (for example, through the use of over provisioning). The streaming client is located in the mobile User Equipment.

Use cases are formed as a combination of QoS-relevant settings and parameters from the items 1-7 above. The PSS characterisation is meant to give insight into how different streaming server and streaming client algorithms and settings in PSS Release 5 perform in the given use cases.

6.1 Multimedia content creation

6.1.1 CBR vs. VBR encoding for video

Rate control strategies for video coding can be classified into constant bit rate (CBR) and variable bit rate (VBR).

The main application of CBR rate control is encoding for transmission over constant rate links (e.g. ISDN) under strict end-to-end delay constraints. Conversational multimedia services, such as video telephony (e.g. 3G-324M) typically employs CBR rate control. The low delay constraint of such applications requires the encoder rate control to generate a video bitstream which when transmitted at the constant channel rate can be decoded and displayed at the receiver virtually without any pre-decoder or post-decoder buffering. In this scenario, the frame selection algorithm of the CBR rate control (i.e. which input frames to encode from the source) is directly driven by the bit-allocation decision of the algorithm. The codec rate control has to ensure that the next frame is not taken from the source before all bits of an encoded frame are transmitted at the constant channel rate. Due to the variable rate nature of video compression, bit-allocation can not in general be kept constant through all frames of the video sequence, thus CBR rate control algorithms inherently generate a not constant picture rate video. In the attempt of still trying to maintain as constant picture rate as possible, CBR rate controls try to limit the number of bits, which can be used for compressing each picture in a video sequence, regardless of how "difficult" it is to compress the picture. The final quality of the compressed video stream, therefore, mainly depends on the complexity of the content (e.g. how difficult it is to compress the content). However, different scenes have different coding complexity. For instance, it is easier to encode a news speaker in front of a fixed background than a soccer game. The coding complexity of a scene is determined by the overall amount of motion and also by the level of detail in each particular picture. CBR coding for video works fine, as long as the complexity of the scene is more or less constant as it is the case for head-and-shoulder scenes with little motion. However, CBR coding of arbitrary video sequences containing scenes with varying coding complexity gives a fluctuating quality and varying frame rate, which has a negative impact on the subjective quality.

VBR video rate control strategies can be used if either the low-delay or the constant transmission rate constraint of the application is relaxed. VBR allows video bitrate variation (i.e. the number of bytes decoded per a defined period can vary over different periods of the stream) and the rate control algorithm is therefore less restricted in the bit-allocation and frame selection. VBR video in general can provide more consistent visual quality by restricting less the inherent variable rate nature of video compression. The variation of bit rate can be still controlled to adhere the channel throughput limitations and pre-decoder and post-decoder buffering constraints of the receiver. Examples and comparison of different rate control methods will be given in section 7.

6.2 Streaming server media transmission

6.2.1 Transmission of VBR content over constant rate channels

Real-time transmission of a variable rate encoded video stream would require a transport channel, which can fulfil at each point in time the streams variable rate demand. However, many typically used Internet access channels are characterized by a certain bottleneck link rate, which cannot be exceeded (e.g. analogue modem speeds, ISDN, and so on). A UMTS WCDMA bearer with strict QoS guarantees is another example for such a bottleneck link. Therefore, rate-smoothing techniques are required which allow streaming of variable rate encoded content at a constant transmission rate [8].

Transmission of variable rate encoded video content over UMTS is explained in Figure 1. The encoder generates variable rate encoded video streams. The transmission rate of the streaming server is adjusted to the available bandwidth on the UMTS streaming bearer, in the example this is a constant rate, which corresponds to the negotiated

guaranteed bitrate. Delivery over UMTS introduces a certain delay jitter, which needs to be compensated for at the streaming client in the de-jitter buffer. In addition to delay jitter compensation, the streaming client buffer is to compensate for the accumulated video encoding rate and transmission rate difference (i.e. pre-decoder buffer). The video buffering verifier of [3] is assumed to be followed by the streaming server.

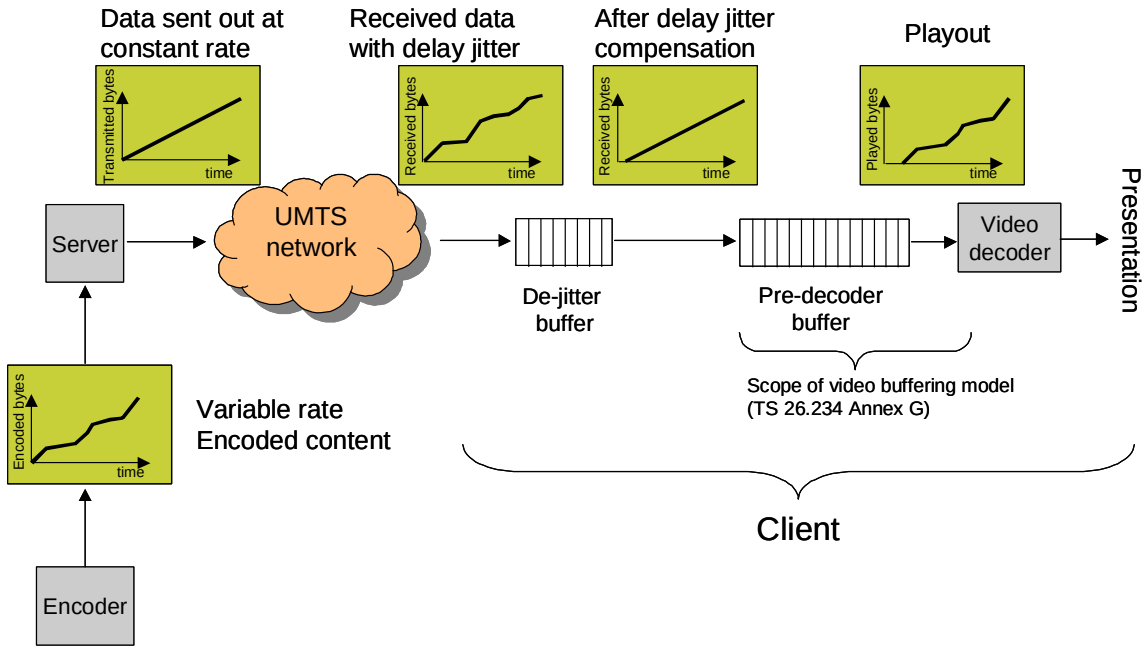

Figure 1: Transport of VBR streams over UMTS

6.2.2 Transport and Transmission

Media streams can be packetized using different strategies. For example, video encoded data could be encapsulated using

- One slice of a target size per RTP packet
- One GOB (row of macroblocks) per RTP packet
- One frame per RTP packet.

Speech data could be encapsulated using an arbitrary (but reasonable) number of speech frames per RTP packet, and using bit- or byte alignment, along with options such as interleaving.

Transmission of RTP packets can occur in different fashions. There are at least two possible ways of making transmission:

- VBRP (Variable Bit Rate Packet) transmission: the transmission time of a packet depends solely on the timestamp of the video frame the packet belongs to. Therefore, the video rate variation is directly reflected to the channel.
- CBRP (Constant Bit Rate Packet) transmission: the delay between sending consecutive packets is continuously adjusted to maintain a near constant rate.

Examples of traffic characteristics for different packetization and transmission techniques are included in section 7.

6.2.3 Packet Sizes

While there are no theoretical limitations for the usage of small packet sizes, implementers must be aware of the implications of using too small RTP packets. The usage of such kind of packets would produce three drawbacks:

1. The RTP/UDP/IP packet header overhead becomes too large compared to the media data;

2. The bandwidth requirement for the bearer allocation increases, for a given media bit rate;

3. The packet rate increases considerably, producing challenging situations for server, network and mobile client.

As an example, Figure 2 shows a chart with the bandwidth repartition among RTP payload media data and RTP/UDP/IP headers for different RTP payload sizes. The example assumes IPv4. The space occupied by RTP payload headers is considered to be included in the RTP payload. The smallest RTP payload sizes (14, 32 and 61 bytes) are examples related to minimum payload sizes for AMR at 4.75 kbps, 12.20 kbps and for AMR-WB at 23.85 kbps (1 speech frame per packet). As Figure 2 shows, too small packet sizes (<= 100 bytes) yield an RTP/UDP/IPv4 header overhead from 29 to 74%. When using large packets (>= 750 bytes) the header overhead is 3 to 5%.

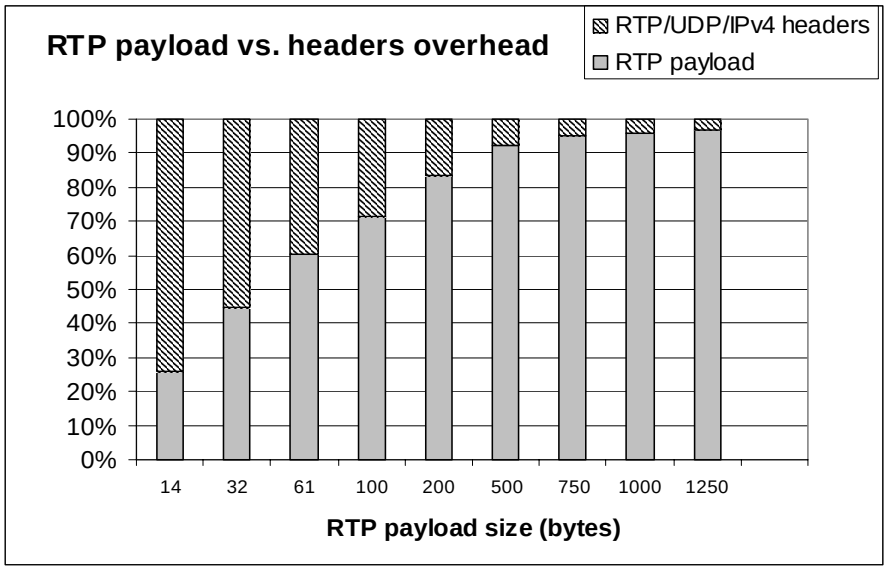

Figure 2. Repartition of bandwidth among RTP payload and RTP/UDP/IP header for different packet sizes

Implementers should also be aware of the implications of using large packets, and of the opportunity of setting limits for maximum packet sizes generated by PSS servers. In general it must be assumed that the larger the payload sizes the higher is the end-to-end latency for the reception of the packets at the PSS client. In case of usage of non-transparent layer 2 protocols, the retransmission procedure introduces an increasing delay jitter for increasing packet sizes for a given Layer-2 loss rate. This happens because the larger the IP packets, the larger is the number of layer-2 blocks subject to individual loss (if there are N layer-2 blocks, N>1, there is the chance of need to retransmit 0 to N layer-2 blocks, yielding a variable delay as N gets larger).

Fragmentation is one reason for limiting packet sizes. It is well known that fragmentation causes

- increased bandwidth requirement, due to additional header(s) overhead;
- increased delay, because of operations of segmentation and re-assembly.

Implementers should consider avoiding/preventing fragmentation at any link of the transmission path from the streaming server to the streaming client, whenever possible and controllable by the PSS server.

Example 1 (IPv4 in the CN):

IPv4 packet size = 1501 bytes

MTU size for IPv4 is the maximum IP packet size before fragmentation = 1500 bytes.

If a PSS server generates packets as above, every packet is split into 2 packets: one 1500 bytes long, and the second 28 bytes long (20 bytes for IPv6 header, and 8 bytes is the minimum fragment size at IP level). So, the transmission of 1501 bytes would require a total of 1500+28=1528 bytes, or about 2% more bandwidth requirement, double IP packet rate and a potential increase (up to double) in packet loss rate.

Over the Iu-ps interface 1400 byte will avoid fragmentation. This is a conservative value to accommodate the protocol layer header overheads. The possible overheads over the Iu-ps interface (GTP/UDP/lower-IP) are the following:

GTP main header = 12 bytes

GTP extension header = 4 bytes

UDP header = 8 bytes

IPv4 header = 20 bytes (without optional IPv4 fields), or

IPv6 header = 40 bytes (without optional IPv6 headers).

The maximum headers size is then 12+4+8+40=64 bytes. The MTU for IPv4 and IPv6 is 1500 bytes. So, the maximum SDU size would be 1500-64=1436 bytes. 1400 bytes is a safer value.

Over the GERAN Gb interface the default size for LLC data field (=SNDCP frame) is 500 bytes in unacknowledged mode LLC. The LLC data field size can be set to a value up to 1520 bytes through explicit request of the MS as is specified in [3]. SNDCP fragmentation of packets larger than 500 bytes is avoided if the mobile station sets the LLC data field size to an appropriate, larger value. The same service can be supported over the Iu and Gb interfaces if the LLC data field size is set to at least 1404 bytes.

Example 2 (GERAN A/Gb unacknowledged SNDCP with default size of LLC data field):

IP packet size = 497 bytes

Maximum IP packet size before SNDCP fragmentation = 500 (default N201-U field in LLC header)- 4 (SNDCP header)= 496.

If a PSS server generates IP packets as above, every IP packet is split into 2 SNDCP packets: one 500 bytes long, and the second 5 bytes long (4 bytes for SNDCP header and 1 bytes data). So, the transmission of 497 bytes would require 500+5=505 bytes, or about 1% more bandwidth requirement and double IP packet loss rate. If a 1500 bytes packet needs to be transmitted with the same limitations, it would generate 4 SNDCP packets, and a total of 1516 bytes (1% extra header overhead), and the IP packet loss rate would be increased by a factor of 4.

When ROHC (Robust Header Compression) [12] is not used in the PDCP layer [11], the application header lengths are:

RTP header = 12 bytes

UDP header = 8 bytes

IPv4 header = 20 bytes (without optional IPv4 fields), or

IPv6 header = 40 bytes (without optional IPv6 headers).

The maximum RTP payload size is then 1400-12-8-40=1340 bytes (including payload headers) for IPv6, and 1400-12-8-20=1360 bytes (including payload headers) for IPv4. This figure is valid for both the Iu and the Gb interface (see note about Gb above).

6.2.4 Adaptation capability

PSS servers can have different levels of adaptability to varying network conditions. A simple classification could be made:

- Simple transmission of a single pre-encoded bitstream: The server can only send a pre-encoded bitstream at its designated target bit rate. The server does not react upon and rely on any feedback from the streaming client.
- Adaptive transmission of pre-encoded bitstreams (advanced adaptation capability): The server can adjust the transmission rate according to feedback from the streaming client. The server can also change other application traffic characteristics, such as changing the packet size or perform stream switching, according to the characteristics of the network.

6.2.5 Clarification of using PSS Video Buffering Verifier in a rate adaptive service environment

This section is meant to establish a better understanding of how the PSS Rel-5 [3] Video Buffering Verifier (Annex G) can be used in practice as a vehicle to provide functional interoperability between clients and servers in a rate adaptive service environment.

6.2.5.1 Clarification of terms and concepts

In the following discussions bitrate control will be described with reference to the bitrate evolution plots (i.e. sampling curve, transmission curve, reception curve, playout curve), and the term "curve control" will be used in place of rate control.

Figure 3A indicates the points where the different curves can be observed in a simplified streaming model.

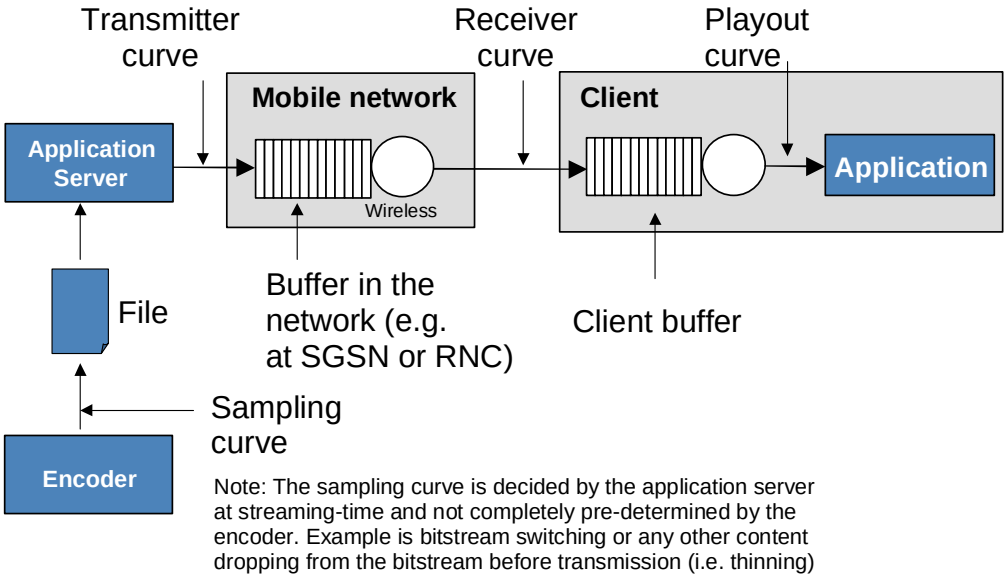

Figure 3A: Illustration of the curves in a simplified streaming model

Figure 3B shows an example bitrate evolution plot. The horizontal axis in the graphs denotes time in seconds; the vertical axis denotes cumulative amount of data in bits. The playout curve shows the cumulative amount of data that the decoder has processed by a given time from the receiver buffer. The sampling curve indicates the progress of data generation if the media encoder was run real-time (it is the counterpart of the playout curve, and is actually a time shifted version of it). The transmission curve shows the cumulative amount of data sent out by the server at a given time. The reception curve shows the cumulative amount of data received and placed into the client buffer at a given time.

The distance between two curves at a given time shows the amount of data between two observation points in the streaming system. For example the distance between the transmission and reception curves corresponds to the amount of data in the network buffer and the distance between the reception and playout curves corresponds to the amount of data in the client buffer. See these examples marked in Figure 3B.

The curve control will be constrained by some limits on the distance between two curves (e.g. max amount of data, or max delay).

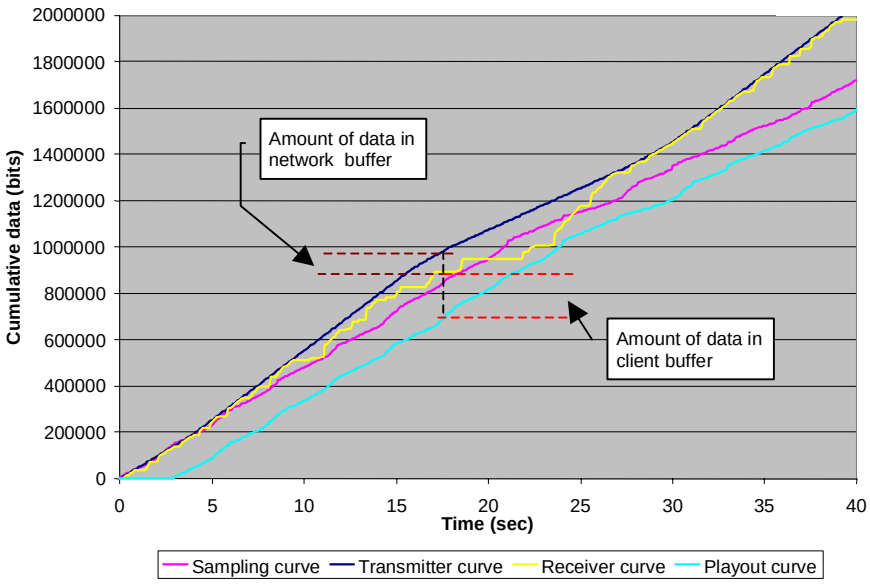

Figure 3B: Example bitrate evolution plot

Term definitions:

1. Pre-decoder buffer = reception curve-playout curve

   "Pre-decoder buffer" refers to the actual pre-decoder buffer at streaming time. "Hypothetical pre-decoder buffer" refers to the pre-decoder buffer as assumed in the hypothetical buffering model. "Pre-decoder buffer size" and "initial pre-decoder buffering period" are parameters of the hypothetical pre-decoder buffer.

   In the hypothetical buffering model, zero delay network and a playout curve exactly following the buffering model (i.e. synchronised) is assumed.

   Zero delay network means that the reception curve is assumed to equal the transmission curve.

   Playout curve exactly following the buffering model means that the sampling curve is assumed to be equal to the playout curve but shifted left by initial pre-decoder buffering period.

   The hypothetical pre-decoder buffer can be traced at streaming time as the difference between the sampling curve-transmission curve. Thus a server controlling the sampling curve-transmission curve effectively controls the hypothetical pre-decoder buffer.

2. Jitter buffering

   The extra pre-decoder buffering required in an actual client, which is to tolerate for packet transfer delay variation (i.e. the maximum expected difference between transmission curve-reception curve).

   PSS client implementations may not include a separate jitter buffer, but jitter buffering is only a function performed by the pre-decoder buffer.

6.2.5.2 Clarification of Annex G buffering parameters

If there is no bitstream switching or other rate adaptation action foreseen, the hypothetical pre-decoder buffer parameters are actually inherent to the bitstream and its transmission schedule (i.e. when each packet is to be sent). These parameters can simply be calculated from the bitstream or were actually used as constraints already at encoding time. It is easy to see how the Annex G video buffering model replaces the MPEG-4 VBV and H.263 HRD in this case.

In case there is bitstream switching or other rate adaptation action foreseen, the server signalled pre-decoder buffer parameters are to be interpreted as the limits to what the server will constrain its difference between the sampling curve and transmission curve during the session. In practice the same pre-decoder buffering model can be followed in a rate adaptation service model, but with a different interpretation of how the server can comply to it.

6.2.5.2.1 What is mandatory?

Whenever the server signals initial pre-decoder buffering period and pre-decoder buffer size parameters the difference between the sampling and transmission curve has to fit into the buffer defined with these parameters (i.e. there is no violation).

This is true regardless whether a predetermined transmission schedule or adapted transmission schedule is used. The rate adaptation must be transparent to this requirement.

6.2.5.2.2 Adaptive transmission curve-reception curve control

In addition to the mandatory sampling curve-transmission curve control, the server attempts transmission curve-reception curve control in order to limit the packet transfer delays (i.e. limit the jitter buffering required at the client).

The variable bitrate over time on the transmission path, and thus variable packet transfer delays, creates the need for transmission curve adaptation.

Unknown future packet transfer delays make it hard for the server to control the transmission curve-reception curve difference.

6.2.5.2.3 Why is it important to have a strict conformance point at the sampling curve-transmission curve control?

The same arguments apply as for the normative definition of MPEG-4 VBV and H.263 HRD -> bitstream/server conformance validation.

The pre-decoder buffer can be implemented at the client as a “static” decoder buffering algorithm that is designed to be conformant to MPEG-4 VBV and is built into the codec (e.g. a DSP SW codec or hardware codec). Such application independent codec conformance implementation is a way to maintain modularity and ensure interoperability between different application modules.

The sampling curve-transmission curve control algorithm can work independently of the transmission curve-reception curve control algorithm, thus it can be implemented on top of any “standard” congestion control algorithm (i.e. transmission curve-reception curve control) such as the IETF defined TCP Friendly Rate Control (TFRC).

6.2.5.3 The resulting constraints and responsibilities

By placing only sampling curve-transmission curve control requirements on the server, any parameter that is not controllable directly by the server is excluded. There is no uncertain or estimated parameter used in this curve control.

There is no indication of preference about the transmission curve-reception curve control in either the server to client or client to server direction. It is completely up to the server to manage it and up to the client to adapt its jitter buffering to the resulting reception curve.

Thus, in practice to ensure stability and minimal functional interoperability, the server will probably take a conservative approach, and try to minimise the transmission curve-reception curve difference at all times (i.e. reception curve = transmission curve).

6.2.5.4 Example scenario relying on 3GPP QoS guarantees

A streaming session setup scenario comprising the following steps is an example of how the different buffering and rate control related parameters can be interpreted and applied in a rate adaptive service environment.

1. Offline encoding of a set of bitstreams at different bitrates. The bitrate range should be around the highest bitrate allowed by the codec level in use in PSS, but should also include lower and higher bitrate streams. Each of which bitstreams together with its transmission schedule is conformant to the hypothetical pre-decoder buffering model with the default parameters (or close to it).

2. Client sends to the server in the capability exchange process a pre-decoder buffer size parameter which is close to its maximum pre-decoder buffer size.

3. Using the given bitstream set (i.e. I-frame placement and stream bitrate) and assuming a given worst case transmission rate adaptation sequence (assuming a pre-defined transmission curve-reception curve control

strategy and worst case reception rate variation), server estimates whether it can guarantee without significant quality loss a maximum sampling curve-transmission curve difference smaller than or equal to the client signalled parameters. It can also decide to not commit to the client signalled parameters, but require higher values than that. This algorithm also outputs a safe recommended initial pre-decoder buffering period to be applied for the bitstream set.

4. Server sends an SDP using the average bitrate stream bitrate and the pre-decoder buffer parameters (i.e. max difference between the sampling and the transmission curve) that it attempts to guarantee.

5. Client requests a streaming RAB with QoS parameters similar to those in Annex J of TS26.234 [3].

6. Client analyses the granted QoS parameters by the network and decides how much jitter buffering there needs to be. In case of strict QoS scheduling on the network, the maximum expected time difference between transmission curve and reception curve is in fact the granted “transfer delay” QoS parameter.

7. Client decides whether it can accept the server signalled parameters (i.e. whether the sum of the server signalled pre-decoder buffer size and buffer size required for jitter buffering exceeds some hard limit of the client pre-decoder buffer size). It can decide not to continue with the session setup if it can not provide the required pre-decoder buffer, and can release the streaming bearer.

8. Client sets up a total pre-decoder buffer size as the sum of server signalled pre-decoder buffer size (i.e. maximum sampling curve-transmission curve difference) and estimated maximum transmission curve-reception curve difference.

9. Client sends a SETUP request and waits for the OK from the server.

10. The client sends a PLAY request, the server responds OK and starts streaming.

11. Client pre-rolls into the pre-decoder buffer for a time which is the sum of initial pre-decoder buffering period and the maximum transfer delay.

12. The server will operate the sampling curve-transmission curve control with the parameters that it signalled.

13. The server will be responsible to explore the max transfer delay limit of the network, and operate its transmission curve-reception curve control to avoid packet drops by the network due to enforcing of the max transfer delay.

6.3 UMTS QoS profile parameters

The UMTS QoS profile [4] is used as the interface for negotiating the application and network QoS parameters. In the following some PSS application specific interpretation of the QoS profile parameters is given. The shown PSS performance in the use cases should be achievable when the only knowledge available about the streaming bearer before starting the streaming session is the knowledge extracted through the following interpretation of the QoS parameters.

6.3.1 Guaranteed and maximum bitrate

The guaranteed bitrate can be understood as the throughput that the network tries to guarantee.

The maximum bitrate is used for policing in the core network (i.e. at the GGSN). Policing function enforces the traffic of the PDP contexts to be compliant with the negotiated resources. If downlink traffic for a single PDP context exceeds the agreed maximum bit rate, user IP packets are discarded to maintain traffic within allowed limits. IP packets could additionally be discarded at any bit rate between the guaranteed and the maximum, when enough resources are not available for the PDP context.

In case of a streaming application, it is possible to shape the excessive traffic and queue those packets exceeding the guaranteed bitrate since the application buffer relaxes the delay requirements. This queuing consists of scheduling packets from a connection up to the maximum throughput and the rest of the packets remain in the corresponding queue.

6.3.2 SDU error ratio

This is the target average SDU error ratio that the network attempts to keep all the time. In some instants this error ratio could be higher than the average target, but an upper bound cannot be defined. The SDU error ratio is computed above the RLC layer.

6.3.3 Residual bit error rate

This is the target average residual bit error rate that the network attempts to keep all the time. In some instants this error ratio could be higher than the average target, but an upper bound cannot be defined.

6.3.4 Maximum SDU size

To guarantee a given SDU error ratio, the larger the SDU size, the smaller RLC BLER the radio interface has to provide, which means that the reliability requirements for the radio link are more stringent. Maximum SDU size should be commonly considered with the required SDU error ratio. From the network viewpoint, smaller SDUs allow easier compliance to reliability requirements by relaxing the radio link adaptation. The application should always be conservative when specifying a maximum SDU size, and set the maximum SDU size parameter to be larger than the maximum expected RTP packet size (plus UDP/IP overhead) (see section 6.2.3). 1400 bytes for the maximum SDU size is a safe value.

6.4 Bearer and Layer 2 network protocols options

6.4.1 UTRAN streaming bearer implementation options

The most critical quality of service limitations in the UMTS network are at the RAN. The details and dynamics of the physical layer is not discussed, only layer-2 and higher implementation options. The listed options for streaming bearer implementation are not meant to be exhaustive, but only meant to show that alternatives for the implementation exist. The network model is constructed based on these mentioned alternatives. In an implementation other not mentioned options and algorithms might be used. The streaming service should actually work independently from the bearer implementation details, as stated in the PSS service requirements [6]. In the following, RLC SDU means a packet in input to the RLC transmitting entity and in output from the RLC receiving entity. RLC PDU means a packet in output from the RLC transmitting entity and in input to the RLC receiving entity. These definitions are given according to [7].

6.4.1.1 UTRAN RLC modes

There are three different traffic handling modes in UTRAN radio link layer (i.e. RLC) for transporting user-plane data: Transparent Mode, Unacknowledged Mode and Acknowledged Mode.

The transparent mode passes RLC SDUs without additional header information through. No SDU concatenation or padding is possible. The transparent mode is primarily targeted to be used with circuit switched bearers. In a packet switched bearer, transparent mode is useful if the RLC SDU size is adapted to the RLC PDU size. In a general video (and some audio) stream, size of packets will vary and it can not always be an integer multiple of the size of an RLC-PDU. Therefore the transparent mode is not recommended to be used with the streaming traffic class.

The unacknowledged mode introduces a more flexible RLC SDU mapping to RLC PDUs, and thereby makes it suitable for general packet based traffic.

Transparent and unacknowledged mode L2 bearers normally carry delay sensitive traffic, as there is no delay introduced for error detection and correction.

The acknowledged mode provides error correction by applying re-transmission for erroneously received RLC blocks. As the acknowledged mode provides in-order delivery of SDUs, enabling the retransmission scheme results in added delay for SDUs whose RLC blocks are being re-transmitted. This appears as SDU delay jitter at the receiver.

The retransmission is not guaranteed to provide full reliability. Any yet unacknowledged RLC block may be discarded from a sender retransmission buffer (i.e. the retransmission attempts for that block stopped) if one of the following occurs: timer expiration, maximum number of retransmission attempts reached or sender retransmission buffer overflow.

This means, that RLC acknowledged mode can be flexibly configured to trade off the required reliability and maximum delay allowed in the RLC layer.

6.4.1.2 Implications of RLC mode decision

A PSS application can tolerate startup delays of multiple seconds (e.g. 2-4 seconds), thus can implement long delay jitter buffers. This implies that PSS applications are not overly sensitive to network delay jitter. In addition to that, streaming applications, particularly video, are much more sensitive to packet loss than delay jitter. It gives a worse viewing experience to see some video picture data missing, than having some video picture displayed late.

Therefore, despite the high delay jitter introduced by using RLC acknowledged mode (AM), it is possible to use RLC retransmission for correcting damaged RLC blocks instead of reflecting directly the RLC loss up to the application.

Typically the radio link is adapted in UTRAN by transmission power (in GERAN by selection of coding schemes). Instead of relying on high transmission power (or protective coding scheme) in order to achieve a given SDU error ratio as requested by a given QoS profile, RLC re-transmissions can be used. It makes the implementation of the streaming bearer in the network cheaper at the expense of possibly introducing higher delay jitter.

6.4.1.3 Examples of bearers for PSS

Bearers for PSS should take into account two types of traffic:

- RTSP traffic for session control
- HTTP/TCP traffic for SMIL presentations and still images, bitmap graphics, vector graphics, text, timed text, and synthetic audio
- RTP and RTCP media and control traffic.

RTSP and HTTP traffic would need for example an interactive bearer at 8/16/32 kbps for downlink and uplink. RTP and RTCP traffic would be, for example, carried over bearers of 16/32/64/128 kbps in downlink and 8/16 kbps in uplink.

Further information about the possible bearers for PSS is available in [9] [10].

6.4.2 GERAN streaming bearer implementation options

6.4.2.1 Iu and A/Gb modes

In GERAN the GSM/GPRS/EDGE radio technology is utilised. The GERAN is, from Release 97 and onwards, connected via the Gb interface to the 2G PS CN. From Release 5 and onwards GERAN also supports the Iu interface to the 3G PS and CS CN. Mobile stations using the Gb interface are said to operate in A/Gb mode and mobile stations using the Iu interface operate in Iu mode.

In A/Gb mode the SNDCP/LLC protocols are used in the 2G-SGSN. SNDCP and LLC protocols provide unacknowledged and acknowledged services.

In Iu mode the PDCP protocol located in the RAN is used. The PDCP protocol in GERAN Iu mode is exactly the same as PDCP in UTRAN.

Both Iu mode and A/Gb mode use an RLC/MAC protocol located in the RAN. The RLC/MAC protocol of GERAN Iu is built using the RLC/MAC protocol of A/Gb mode and includes enhancements to support all UMTS traffic classes.

6.4.2.2 GERAN RLC modes

The GERAN RLC unacknowledged and acknowledged modes are in their operations similar to their UTRAN counterparts. In GERAN L2 retransmission can use Incremental Redundancy (IR). IR refers to a hybrid ARQ scheme, where different channel coding can be used for repeated copies of the same data block, thus enabling combining of the channel decoded original and retransmitted block, which enhances the spectral efficiency of retransmissions.

There is also a difference in how delay bounds are enforced in the scheduler queue. In GERAN, once an RLC block has been transmitted (but not yet acknowledged in RLC acknowledged mode), it can not be discarded from the queue any

more. This means that there is no way to limit the number of retransmission attempts and the RLC-acknowledged mode will always be full-persistent. The “RLC Discard” mechanism is used instead for scheduler queue length management, and to enforce application requested delay bounds for packets. The mechanism discards packets that have exceeded some max time limit for staying in the scheduler queue. The RLC Discard timer has to be tuned to work well with the receiver buffering delays and the scheduler queue thresholds assumed by the rate adaptation scheme in the streaming system.

6.5 Network transport channel mapping

6.5.1 Dedicated or shared channel

In UTRAN several schemes may be considered for channel allocation for streaming traffic class connection (downlink): dedicated channel (only streaming packets are sent through a reserved pipe), shared channel with other non-real time application packets (from the same user or not) or shared channel with other real time packet flows.

One of the latter two cases (i.e. when radio resources are shared among different flows) could be chosen by the RRM for the sake of better network resource utilisation, fairness, statistical multiplexing gain or some other reasons.

When mapping a streaming traffic class RAB to a radio bearer in UTRAN, the following applicable bearer services (transport channels) can be identified:

- DCH (Dedicated Channel) is an up- and downlink channel and is the main transport channel for packet data. DCH is dedicated to one flow and can be used for fairly constant bitrate packet traffic.
- DSCH (Downlink Shared Channel) is a common channel that can be shared among multiple users and multiple flows. DSCH downlink channel is particularly efficient for bursty Non Real Time packet traffic. It is good for asymmetric services, where downlink is the main transmission direction.

It should be noted that the support of DSCH is optional to terminals, therefore there must always be an alternative way to use only DCH, even though the DSCH would be the preferred option.

6.5.2 Implications of channel mapping decision

If a streaming source generates less traffic than its allocated bearer was set-up for, or generates a variable rate traffic, other services could use the unused resources. In this case a shared channel (DSCH) could be used. It is, however, difficult to guarantee QoS to each individual flow competing for the same shared resource. On the other hand, the network wants to make sure, that if a dedicated fixed-rate channel is allocated (DCH) the resource is utilised efficiently by the streaming application. These are the factors driving the choice of transport channel to be used for streaming.

It can be assumed that the effective radio throughput on average will be the same throughout the session independently of the transport channel chosen. Thus the application can assume, that it can transmit at this average radio throughput rate, and the variation of the available radio rate will be hidden behind a large enough scheduler buffer. Similarly, this buffering can also smooth out any temporal variation of the transmission rate around the average rate. Application rate adaptation is necessary when, for any reason this assumption proves not to be valid (e.g. due to different time window sizes used at the network and the application over what the rate is averaged).

The flow mapping decision puts different requirements on the rate adaptation algorithm required. Depending on the expected channel rate variation, a streaming application should be prepared to apply different rate measurement and rate adaptation schemes. Depending on the rate variation model, for example, rate measurements might be interpreted differently. A model of available rate variation in the network, can be built based on the understanding how a streaming bearer with different maximum and guaranteed bitrate QoS parameters is implemented in the network (e.g. mapped to what transport channel).

When a dedicated channel (DCH) with a given bitrate is allocated for the downlink flow, no available rate variation on the air interface is expected. However, if RLC re-transmission is used the rate variation due to retransmission can not always be neglected. The radio channel allocation is usually such, that the expected L2 throughput after re-transmission should reach the guaranteed bit rate.

When streaming is implemented over a shared channel (DSCH), the available bitrate for a single flow varies over time according to some pattern, which depends on many factors e.g. the scheduler algorithm used in the RAN, the load in the cell or some other rate allocation policies. The RRM however aims to maintain on average the guaranteed bitrate.

6.5.3 HSDPA

High Speed Downlink Packet Access (or HSDPA) is part of UTRAN Release 5. With HSDPA, packet scheduling is expected to be very flexible using 2 ms frame size. HSDPA introduces some new features, such as Adaptive Modulation and Coding (AMC) and Hybrid-Automatic Repeat Request (H-ARQ), and scheduling at the Node B. H-ARQ allows retransmissions at layer 1 (between the UE and Node B). This means that PSS could be run over RLC Unacknowledged mode. Without this feature, retransmissions are enabled at layer-2 RLC between RNC and UE. The new HSDPA features allow also to decrease retransmission delays and maximize throughput and peak rates. The very fast retransmission procedures enabled by HSDPA makes this feature suitable for services with variable bit rate and packet sizes, such as variable rate streaming.

6.5.4 EGPRS / GERAN

The EGPRS / GERAN radio physical layer settings will determine the data rate available at the link layer. The data rate depends on the number of allocated time slots within a radio frame to a given mobile (e.g. 3 DL + 1 UL timeslot) and the Modulation and Coding Scheme (MCS) used in the timeslot. MCSs provide that employ a lower code rate can correct more bit-errors, thus are more robust, but provide lower data rates, while less robust MCSs provide higher data rates. The data rate per timeslot can vary from 8.8 kbps (MCS-1) to 59.2 kbps (MCS-9). The instantaneous data rate is computed as combination of the allocated time slots and current MCS used. MCSs can vary during a connection depending on the radio link quality. To guarantee a certain bit rate and/or RLC frame error rate, the network may use a compensation function between allocated time slots and MCSs.

In EGPRS / GERAN radio the concept of dedicated channel (i.e. radio resources dedicated to one given flow only) does not exist. The GPRS capacity (i.e. number of timeslots allocated to packet data) available is to be shared between all mobiles in the system. The resource is to be managed by the packet control unit (PCU) scheduler implemented at the RLC/MAC layer in the RNC. The GPRS capacity is shared by allocating timeslots (i.e. PDTCH channels) according to some ignalling but fair algorithm to the different application packet flows directed to the different mobiles.

6.6 Core network

In this TR it is assumed that no critical problems occur in this segment of the end-to-end PSS chain. In addition, the number of configurations and options for the core network are very large and this analysis is out of the scope of this document.

6.7 Streaming client

PSS clients can have different features and options implemented, such as

- Error concealment tools
- Features of simple PSS client (as defined in Release 4 PSS specifications)
- Features of Extended PSS client (as defined in Release 5 PSS specifications), including pre-decoder buffering

Sending RTCP reports to the PSS server (following Release 4 or Release 5 guidelines).

7 PSS characterisation

7.1 Comparison of different rate control strategies for video streaming

In this section it is assumed that the streaming server has no adaptation capability, and simple transmission of a single pre-encoded bitstream takes place. Video rate control strategies are compared in terms of the achieved subjective picture quality and picture rate when conforming to pre-defined rate variation limits.

Especially for streaming applications, the rate control mechanism described in [8] was proposed. It takes as input a (bottleneck) rate R, an initial buffering delay d and a buffer size s. It then encodes a pre-stored video sequences at

variable rate such that when the stream is transmitted at a constant rate R, it can be played back continuously by a client with pre-decoder buffer size s, after a initial pre-decoder buffering delay d.

In the following, we present some simulation results, which compare the above rate control mechanism for variable rate coding under a certain buffer size limitation with constant rate coding and unconstrained variable rate coding (e.g. unlimited buffer size). Table 1 summarizes the results. A 2 minutes long clip taken from a TV news show was encoded with H.263 at QCIF resolution and 10 frames-per-second. The mean bitrate averaged over the whole stream was in all three cases adjusted to about 50 kbps.

As an objective quality measure, average PSNR values where computed. Higher PSNR usually means better quality, although PSNR values are not always consistent with subjective quality perception. The comparison shows, that unconstrained variable rate coding results in a good quality but also requires the largest buffer size. Constant rate coding requires almost no buffering but the quality of the resulting video is significantly worse compared to variable rate coding. Although the PSNR is 1.5 dB higher, one has to take into account that the constant rate coding control drops complete frames in order to fulfil the strict rate constraint. In the given example a total of 8% of the frames was dropped.

The last row shows the results for the streaming rate control proposed in [8] for an initial buffering delay of two seconds and a maximum buffer size of 20000 bytes. One can clearly see the trade-offs: initial buffering delay and buffer size are according to the pre-specified values, the PSNR is close to the one of variable rate coding. However, no frames were dropped.

Table 1: Comparison between different rate control strategies for a test video sequence

Rate control	Initial buffering [sec]	Buffer size [bytes]	PSNR [dB]
Constant quality / variable rate	0.4	163501	30.8
Constant rate / variable quality (TMN8 rate control)	0.5	6827	32.3, 100 frames (= 8%) skipped
Streaming rate control	1.8	17951	32.0

Figure 3, 4 and 5 give some more detailed insights how the different rate control mechanisms works. Each graph shows three curves, named “Playout”, “MaxBuff” and “Transmission plan”. The horizontal axis denotes time, the vertical axis denotes data counted in bytes. The transmission plan describes how data is sent out by the server. It gives for each time t the amount of data that was sent out by the server. The transmission plan is in all three cases a straight line, which indicates that data is sent at a constant rate (the motivation for constant rate transmission of variable rate encoded video streams is given in the next section). Each Playout curve describe the video data playout behaviour at the client for the different rate control strategies. Since for each point in time the client needs to play out exactly the same amount of data that was generated by the encoder, the playout curve also reflects the rate behaviour of the encoder. The Playout curve denotes the minimum amount of data that a client needs to have received to guarantee smooth playout of the stream. The MaxBuff curve is simply the Playout curve shifted by a certain amount of bytes in vertical direction. The amount if bytes by which this curve is shifted corresponds to the client buffer size. The MaxBuff curve therefore indicates the maximum amount of data that a client may have received without exceeding its buffer.

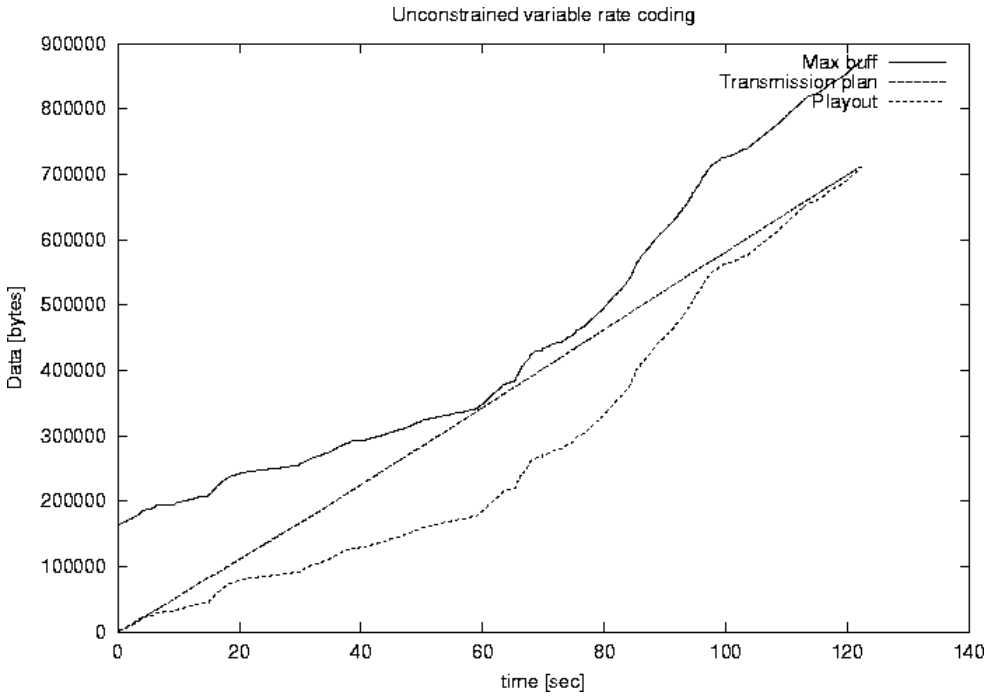

Figure 3: Unconstrained variable rate coding

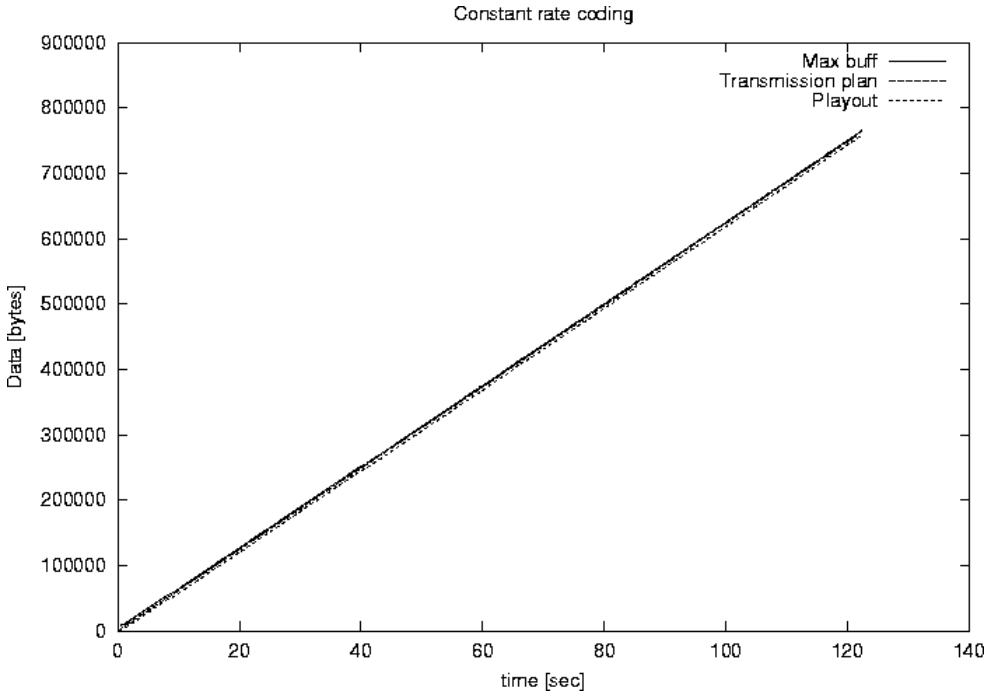

Figure 4: Constant rate coding

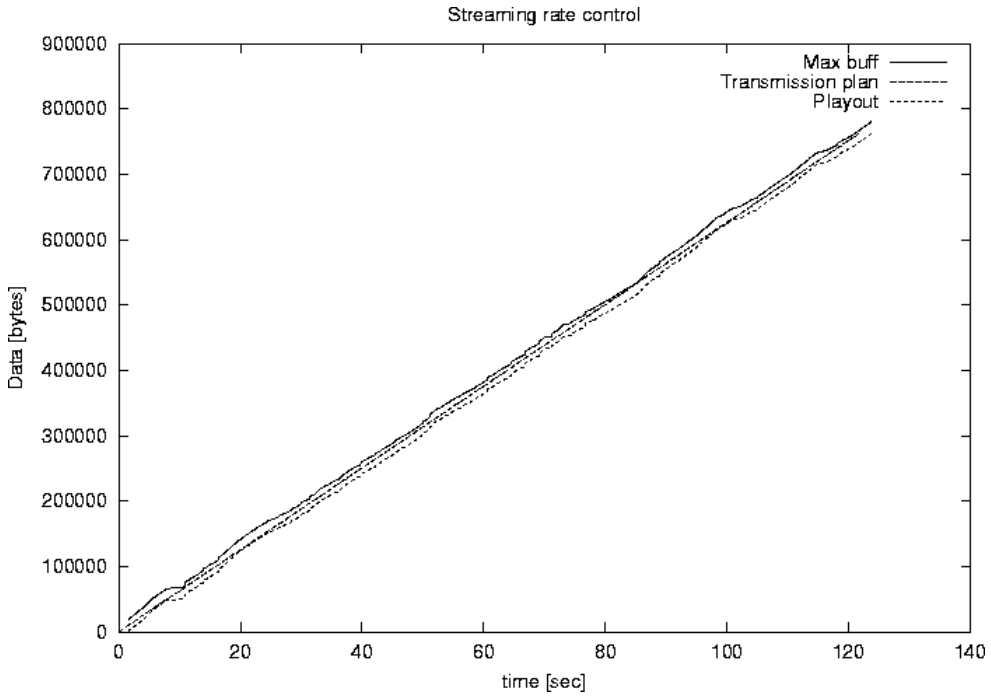

Figure 5: Streaming rate control

Figure 3 shows the result for unconstrained variable rate coding which was achieved by using a fixed quantization parameter for the whole sequence. As one can see, the playout curve differs significantly from the constant rate transmission plan. The maximum distance between the transmission plan and the Playout curve indicates the required buffer size. As can be seen a large buffer size is required in this case. The exact buffer size according to Table 1 is 163501 bytes.

Figure 4 shows the result for constant rate coding. Due to constant rate coding, the rate of the encoded stream is constant and therefore the playout curve is a straight line, which is almost identical to the transmission plan. The required client buffer size in this case is much smaller compared to the previous case.

Finally, Figure 5 shows the different curves for constrained variable rate coding. There is more variation in the playout curve compared to the constant bitrate case but much less compared to the unconstrained variable rate coding case. The required client buffer size in this case is 20000 bytes.

As a conclusion, it can be said, that in general variable rate encoded video streams have a better quality than constant rate encoded streams. The price one has to pay is a certain initial buffering delay and a certain buffer required at the decoder when variable rate encoded video is sent over constant or near constant rate channels. There are special rate control mechanisms, which allow specification of certain buffer limitations, which will then not be exceeded.

7.2 Streaming application traffic characteristics

The purpose of this section is to show how different the traffic characteristics of the packet streams generated by a PSS compliant [3] streaming server can be when different application parameters are used.

A video on demand streaming application use case is assumed without adaptation capability at the streaming server, where a stored pre-encoded video bitstream is transmitted by the streaming server. The traffic characteristics was captured from two streaming servers:

1. A PSS compliant [3] streaming server transmitting an H.263+ Profile 0, Level 10 encoded video bitstream. Server behaviour adaptation based on RTCP feedback was not enabled.

2. Publicly available RealNetworks system (RealProducer Basic streaming encoder, RealServer 8.0 streaming server, RealPlayer 8.0 streaming client). Single stream encoding used, but the RealSystem still uses some server behavior adaptation strategy. This server is ignallin for streaming over the Internet.

Two different setups were used for the streaming server in 1.:

- Variable bitrate packet transmission (VBRP)
- Constant bitrate packet transmission (CBRP)

In case of server 1. different packetization algorithms were tested:

1.I. One frame per RTP packet without maximum packet size limitation

1.II. One GOB (row of Macroblocks) per RTP packet

1.III. A target RTP packet payload size (=600 bits) is maintained by using H.263 Annex K slices

In case of server 1. different video rate control algorithms were used in the H.263+ video encoder:

1.A. Fixed-QP encoding
A fixed constant quantization parameter (QP=10) is used for encoding the whole video sequence, thus the inherent rate variation of the encoded video sequence is actually not modified.

1.B. Rate control designed for video streaming given some pre-decoder buffering constraints [8] (referred also to as StreamRC)
It maintains fixed frame rate and consistent quality by utilising the available pre-decoder buffer at the PSS receiver (as described in Annex G of [3]) and requiring an initial buffering time before starting decoding.

1.C. TMN5 rate control
Not video streaming optimised, but designed for real-time encoded, low-delay communicational applications (such as video conferencing), thus resulting in video frame rate variation.

To show how different network conditions can affect the traffic characteristics when server behavior adaptation based on receiver feedback is used (such as in case of the server 2.), two different networks between the server and client were simulated.

- Perfect LAN with low, near-constant packet transmission delay and no packet loss
- Simulated Layer 2 and 3 of UTRAN with 76.8 Kbps dedicated channel, RLC frame size 640 bits, RLC unacknowledged mode. Layer 1 is not simulated, thus no RLC frame errors are applied. 60 ms RAN delay is assumed both in the uplink and downlink.

In the simulations a video sequence was captured at 15 fps at QCIF (176x144) resolution. The video content of the sequence is a combination of different type of scenes with multiple scene cuts. It includes both fast and slow motion content with sometimes large camera movement and also some almost steady shots in between. It can be considered a typical video on demand streaming sequence.

For a representative video sequence the following statistics is presented:

- average, minimum and maximum packet size and standard deviation of the packet size distribution (the packet size includes RTP/UDP/IP header overhead)
- histogram of used packet sizes
- average, minimum and maximum bitrate (bitrate samples are calculated over non-overlapping 1 second windows as the total number of bytes in packets sent in the window) and standard deviation of the bitrate distribution
- plot of bitrate variation over time

7.2.1 Packet size statistics

1.A (Fixed QP=10) / LAN IP Packet size (bytes)	Average	Standard Deviation	Maximum	Minimum
III (Slice)	106	56	181	45

1.B (LWRC) / LAN IP Packet size (bytes)	Average	Standard Deviation	Maximum	Minimum
I (Frame)	573	398	4303	67
II (GOB)	99	88	663	43
III (Slice)	108	56	210	45

1.C (TMN 5) / LAN IP Packet size (bytes)	Average	Standard Deviation	Maximum	Minimum
I (Frame)	595	229	3375	62
II (GOB)	102	79	759	43
III (Slice)	109	56	241	45

Section 2003. . / LAN IP Packet size (bytes)	Average	Standard Deviation	Maximum	Minimum
N/A	521	154	668	64

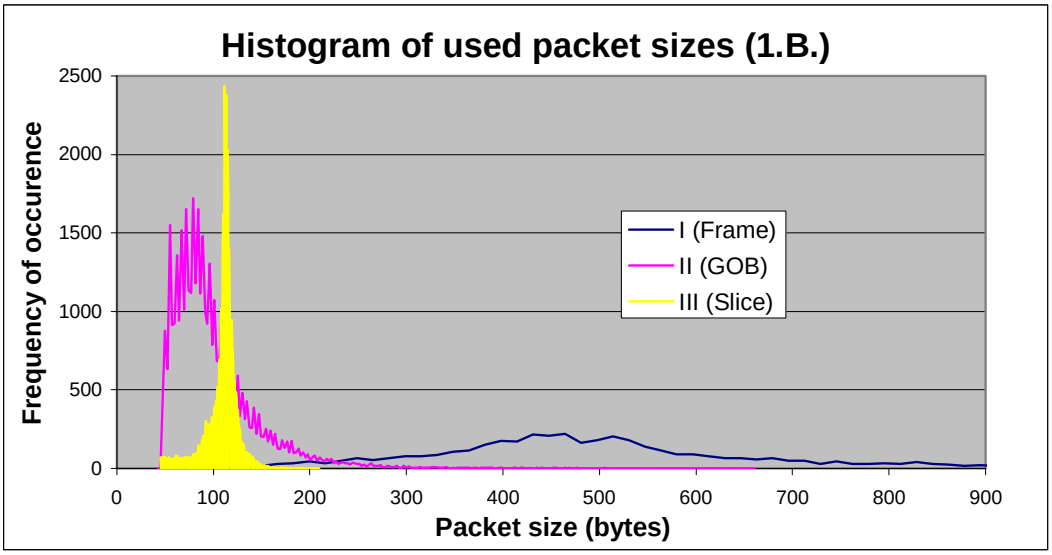

Figure 6 – Packet sizes for different packetization algorithms (LWRC)

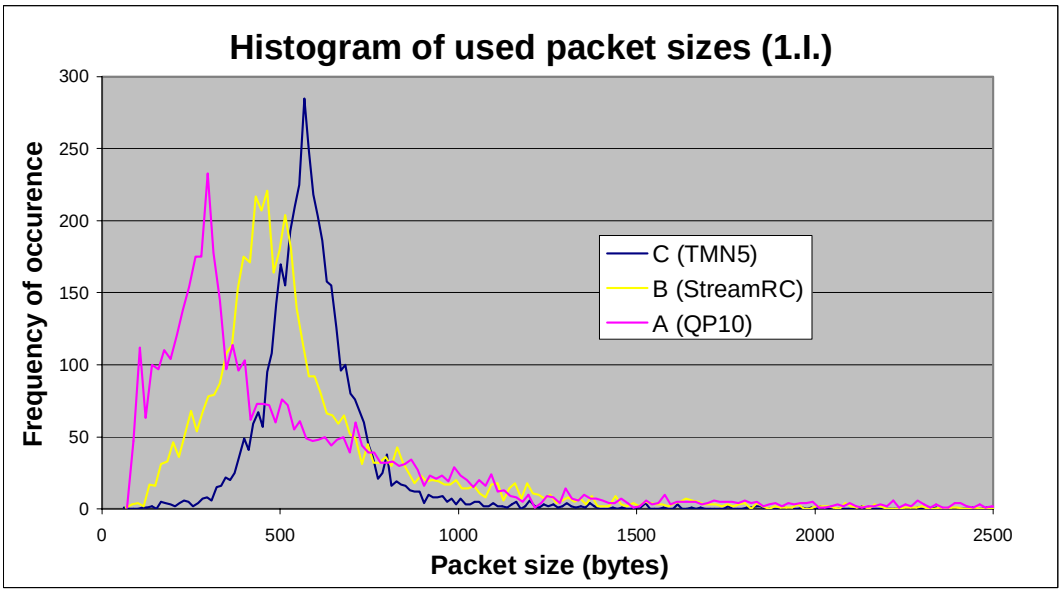

Figure 7 – Packet sizes for different rate control algorithms (1 frame per RTP packet)

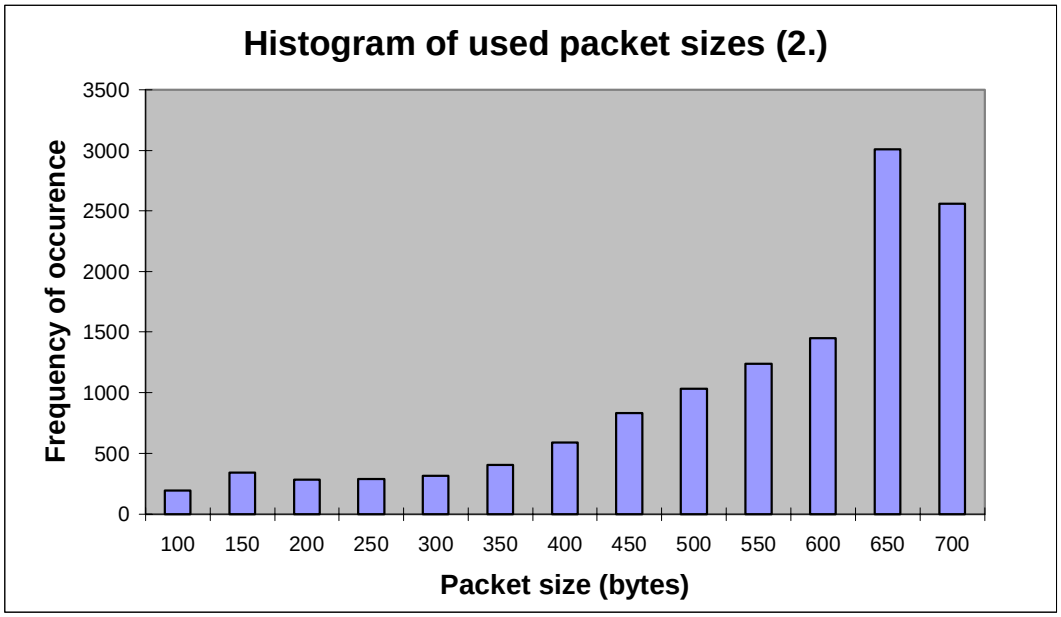

Figure 8 – Packet sizes for Real Networks streaming

7.2.2 Packet Bitrate statistics

1.1.III (VBRP)/LAN Bitrate (bits/s)	Average	Standard Deviation	Maximum	Minimum
A (QP10)	64020	58118	356328	5368
B (StreamRC)	64519	27195	184448	17672
C (TMN5)	63192	1835	71440	54696

1.2.III (CBRP)/ LAN Bitrate (bits/s)	Average	Standard Deviation	Maximum	Minimum
A (QP10)	62913	808	65989	60797
B (StreamRC)	63495	785	66183	61268
C (TMN5)	63522	972	67890	59851

Bitrate (bits/s)	Average	Standard Deviation	Maximum	Minimum
LAN	49282	5010	66061	40898
UTRAN 0% FER	49499	5580	70322	39154

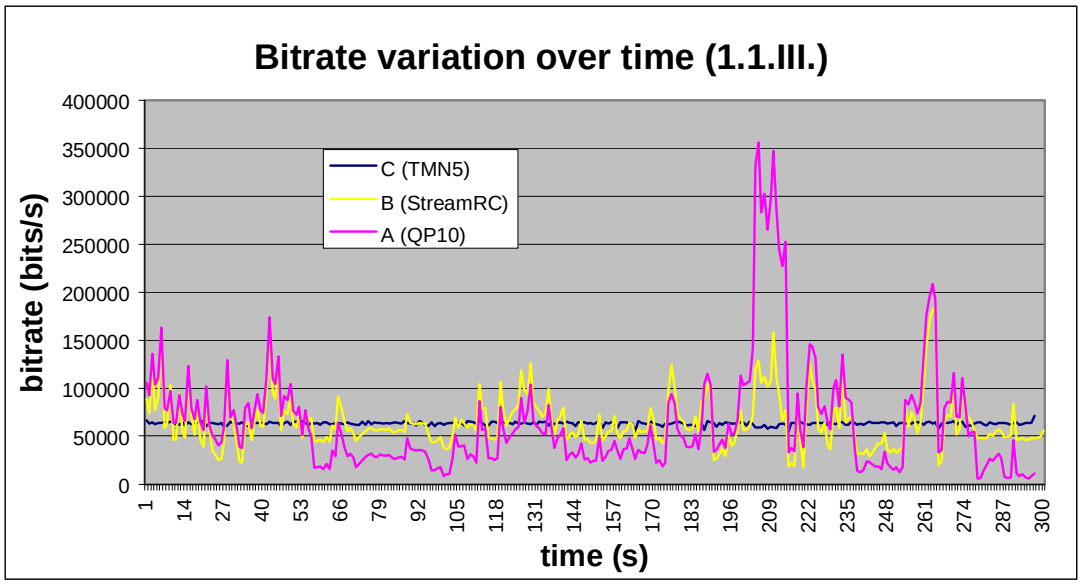

Figure 9 – Bitrate variation for different rate control algorithms (VBRP)

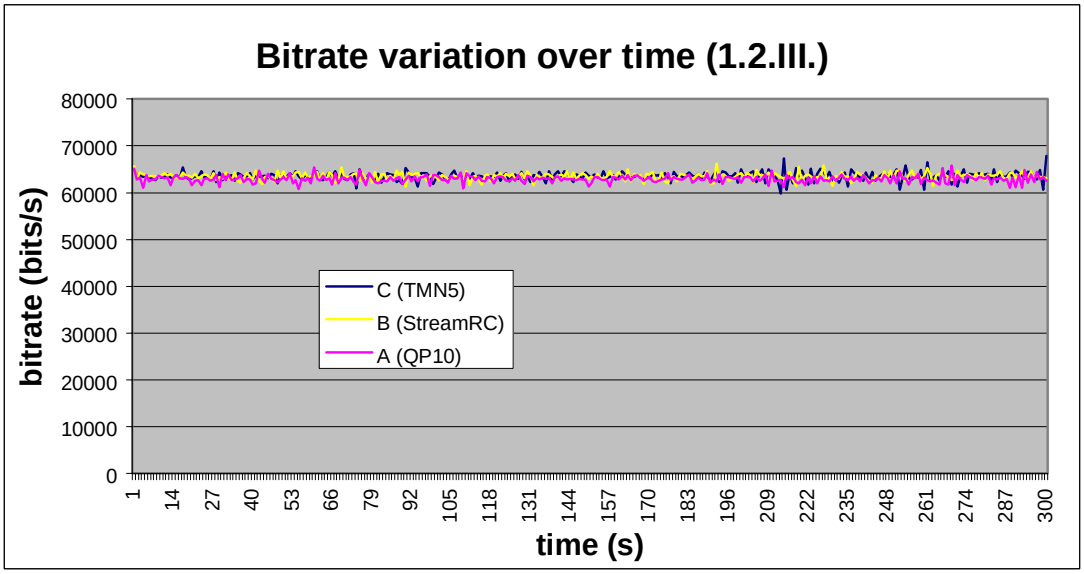

Figure 10 – Bitrate variation for different rate control algorithms (CBRP)

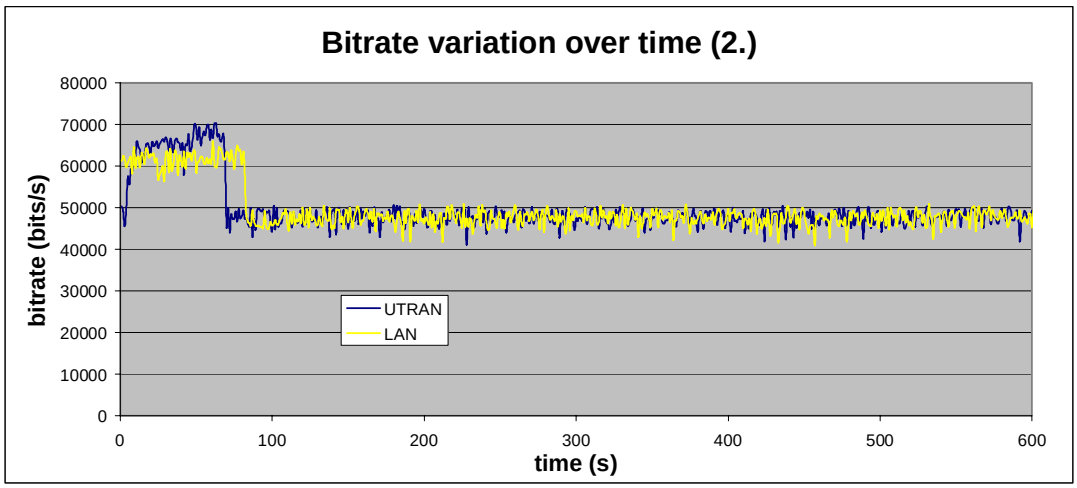

Figure 11 – Bitrate variation for Real Networks streaming over different network scenarios

7.3 UTRAN DCH with RLC Acknowledged Mode

For UTRAN, a Radio Bearer using a dedicated channel and RLC running in acknowledged mode could ignal the requirements of recovering from lost RTP packets and having a fairly stable network throughput behaviour. First of all, a dedicated channel can maintain a fixed transport channel rate on the physical layer. Secondly, when used in acknowledged mode, the probability of lost IP packets is close to zero due to an efficient retransmission protocol on the RLC layer, which retransmits only the erroneous PDUs of an IP packet (note that a PDU corresponds to a small fragment of an IP packet). The increase in IP packet delay jitter caused by this RLC retransmission mechanism is acceptable for streaming services. The WCDMA channel in these tests was emulated by a fairly detailed layer 2 and lower layer protocol implementation. An uncongested cell was also assumed.

Radio Bearer parameters:

- Rate = 64000bps
- TTI = 20ms
- 2 RLC PDUs per TTI
- RLC PDU size: 80 bytes
- 10% block error rate (BLER).

The video sequence was encoded using a constant quantizer (Q=18) and no rate control were used. Only the first frame was encoded in INTRA-mode. No specific INTRA refresh method was employed (the stream contains however a lot of INTRA-coded information due to frequent scene changes). RTP packetization was done at the frame level. SDU size was limited to 1500 bytes. The streaming client buffer size was set to 20000 bytes. The bitrate generated by the streaming server was limited to 58 kbps, about 10% less than the network bit rate to allow retransmission of lost RLC blocks. The maximum number of RLC retransmissions in the RLC Ack-mode was set to be theoretically infinite (persistent retransmission). The average packet size in this example was 628 bytes (including headers).

Figure 12 shows the simulations results. Only the first 15 seconds of the transmission are shown.

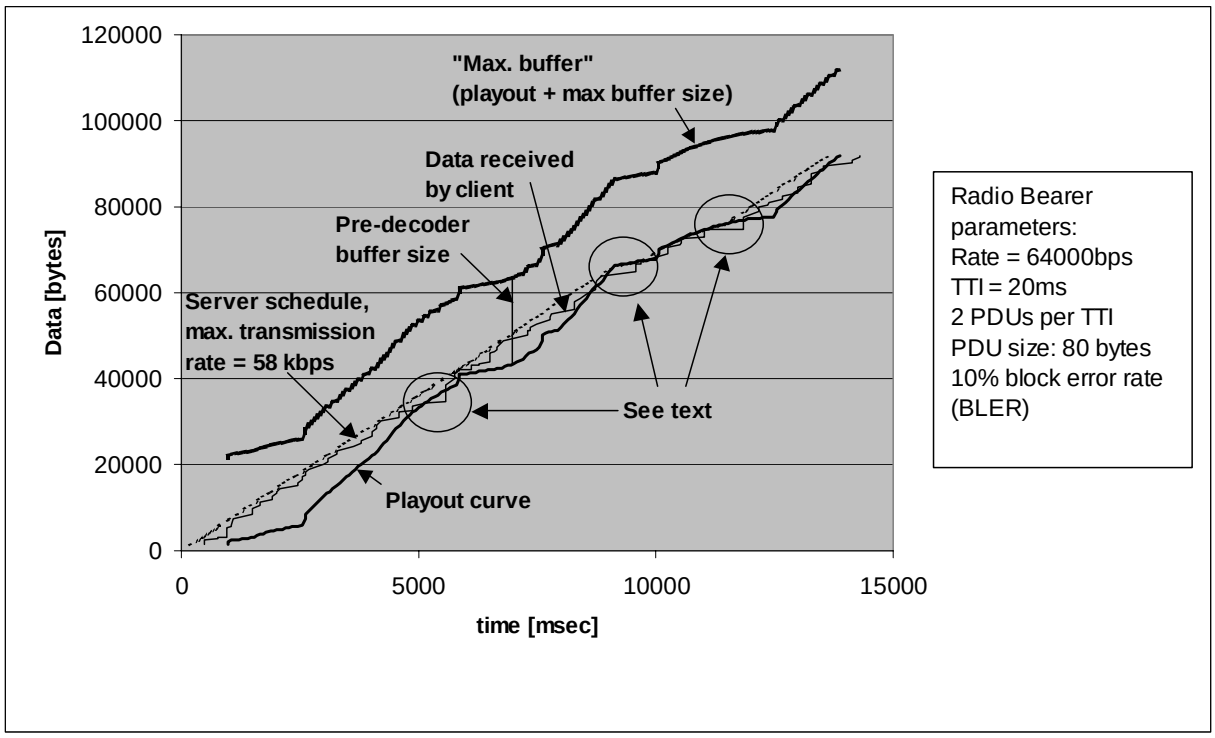

Figure 12: Impact of the delay jitter introduced by a DCH with RLC AM on streaming playout performance

The horizontal axis denotes time in milliseconds; the vertical axis denotes an overall amount of data in bytes. The playout curve shows the minimum amount of data that needs to be available at the decoder for smooth playout. As one can see, playout starts after an initial buffering delay of 1 second, which is needed in this example to play out the stream smoothly.

The “Max buffer” curve represents the maximum amount of bytes that can be stored at the decoder before a buffer overflow occurs. This curve is simply a vertically shifted version of the playout curve. The value by which the curve is shifted represents the client buffer size.

Between the playout and the “max buffer” curve there are two additional curves. The first one represents the amount of data as sent out by the server. The second curve represents the amount of data that is received by the client after transmission over a simulated bearer using RLC AM. Note that the curve representing the amount of data sent out by the server must not cross either the playout or the max buffer curve. Crossing the playout curve would result in a buffer underflow, which leads to a playout interruption. Crossing the “max buffer” curve would result in a buffer overflow, which leads to data losses.

The output stream of the constant quality encoder was smoothed by a traffic smoother. The traffic smoother makes sure that the maximum transmission rate of the video stream is not higher than the maximum channel capacity. Secondly it computes a schedule that minimizes the receiver buffer size by transmitting packets as late as possible (in the literature this is referred to as ‘late scheduling’ in contrast to ‘early scheduling’ where packets are sent as early as possible).

By looking at the amount of data received by the client after transmission over a simulated bearer in acknowledged mode, one can see that the delay jitter introduced by the bearer would lead to buffer underflows. In the example this happens around second 6 and 10. We want to point out that the observed maximum number of RLC retransmissions was less than or equal to 4.

To accommodate for the delay jitter, the playout curve needs to be shifted to the right (= increase in initial buffering delay) by the maximum delay introduced by the bearer. In the given example, this maximum delay was around 1 second. At the same time the buffer needs to be increased by the number of bytes that are transmitted at the maximum transmission rate during 1 second. For a 64 kbps bearer this means 8000 bytes. However, from looking at the curve, one can see that by applying a more intelligent schedule both the additional buffering time and also the additional buffer size could be further reduced. The figure presented here does not consider any further optimisations and therefore reflect a worst-case scenario.

Figure 13 shows the cumulative distribution function (C.D.F.) for the packet delays. As can be seen, in 95% of the cases the delay of a packet is less than one second.

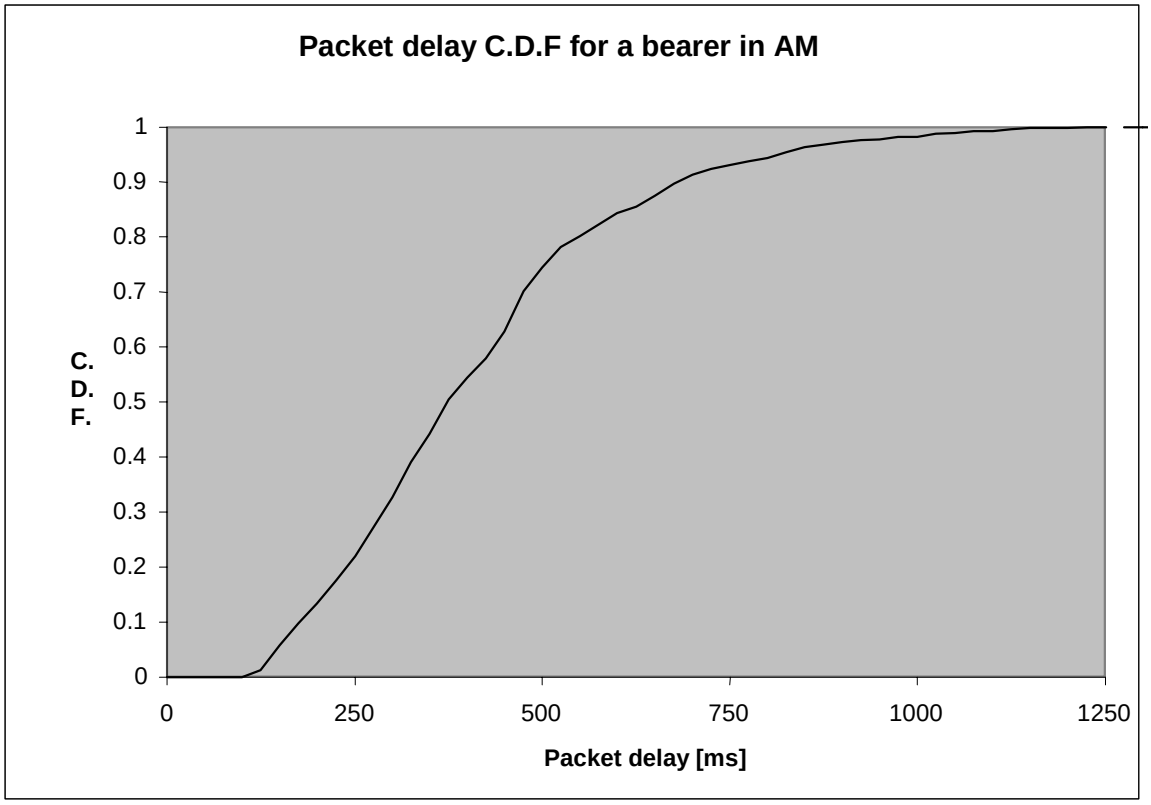

Figure 13: Simulated packet delay C.D.F. for DCH using RLC AM

7.4 Use cases for QoS profile settings

This section contains examples of QoS profile setting for different PSS use cases. In section 6.4.1.3 example bearers for PSS over UTRAN are presented. Here four use cases will be considered, all over a 64 kbps bearer in downlink and a 8 kbps bearer in uplink configured in RLC Acknowledged mode. In the use cases presented, we assume that ROHC is not used. In addition, a use case over GPRS is also considered. Only RTP and RTCP traffic is considered. The use cases ignalli are:

1) Voice only streaming (AMR at 12.2 kbps)
2) High-quality voice/low quality music only streaming (AMR-WB at 23.85 kbps)
3) Music only streaming (AAC at 52 kbps)
4) Voice and video streaming (AMR at 7.95 kbps + video at 44 kbps)
5) Voice and video streaming (AMR at 4.75 kbps + video at 30 kbps) over GPRS

In the parameters for guaranteed and maximum bit rates a granularity of 1 kbps is assumed for bearers up to 64 kbps, as defined in the TS 24.008. Therefore the “Ceiling” function is used for up-rounding fractional values, wherever needed.

During streaming, the packets are encapsulated using RTP/UDP/IP protocols. Here we only consider the IPv4 protocol which leads to the following packet sizes:

IP header: 20 bytes for IPv4 (IPv6 would add a 20 bytes overhead)

UDP header: 8 bytes

RTP header: 12 bytes.

In the following examples, it is assumed that the RS and RR SDP parameters for RTCP bandwidth are assigned values (in bps) corresponding both to 2.5% of the session bandwidth.

The UMTS QoS profile tables of the first four use cases are to be considered instances of the more general QoS profile template described in Annex J of [3].

7.4.1 Voice only AMR streaming QoS profile

Here we are interested in streaming AMR data at 12.2 kbps. We will consider the cases of transmission of 1 and 10 frames per RTP packet. An AMR frame has a length in time of 20 ms, which is between 32 and 35 bytes, depending on the options used (octet-alignment, CRC and interleaving) and including AMR RTP payload header.

Examples:

1 frame per packet: 20 (IPv4) + 8 (UDP) + 12 (RTP) + 35 (max AMR RTP payload) = 75 bytes

10 frames per packet: 20 (IPv4) + 8 (UDP) + 12 (RTP) + 332 (max RTP payload for 10 AMR frames) = 372 bytes.

Table 2: QoS profile for AMR voice streaming at 12.2 kbps

QoS parameter	Parameter value	Comment
Delivery of erroneous SDUs	No	
Delivery order	No	
Traffic class	Streaming	
Maximum SDU size	1400 bytes	
Guaranteed bitrate for downlink	Ceil(30.8)=31 kbps (1 frame/packet) Ceil(15.3)=16 kbps (10 frames/packet)	Including 2.5% for RTCP
Maximum bit rate for downlink	Equal or higher than guaranteed bit rate	
Guaranteed bitrate for uplink	[Ceil(0.12)=1] <= x <= [Ceil(0.8)=1] kbps (1 frame/packet) [Ceil(0.12)=1] <= x <= [Ceil(0.4)=1] kbps (10 frames/packet)	Used for RTCP feedback. The full rate is used for 2.5% feedback. The smaller rate is used for feedback every (at least) 5 seconds.
Maximum bit rate for uplink	Equal or higher than guaranteed bit rate	used for RTCP feedback.
Residual BER	10^{-5}	16 bit CRC
SDU error ratio	10^{-4}	
Traffic handling priority	Subscribed traffic handling priority	not relevant
Transfer delay	2 s	

7.4.2 High quality voice/low quality music AMR-WB streaming QoS profile

Here we are interested in streaming AMR-WB data at 23.85 kbps. We will consider the cases of transmission of 1 and 10 frames per RTP packet. An AMR-WB frame has a length in time of 20 ms, which is between 61 and 64 bytes, depending on the options used (octet-alignment, CRC and interleaving) and including AMR RTP payload header.

Examples:

1 frame per packet: 20 (IPv4) + 8 (UDP) + 12 (RTP) + 64 (max AMR RTP payload) = 104 bytes

10 frames per packet: 20 (IPv4) + 8 (UDP) + 12 (RTP) + 622 (max RTP payload for 10 AMR frames) = 662 bytes.

Table 3: QoS profile for AMR-WB high quality voice/low quality music streaming at 23.85 kbps

QoS parameter	Parameter value	Comment
Delivery of erroneous SDUs	No	
Delivery order	No	
Traffic class	Streaming	
Maximum SDU size	1400 bytes	
Guaranteed bitrate for downlink	Ceil(42.7)=43 kbps (1 frame/packet) Ceil(27.2)=28 kbps (10 frames/packet)	Including 2.5% for RTCP
Maximum bit rate for downlink	Equal or higher than guaranteed bit rate	
Guaranteed bitrate for uplink	[Ceil(0.12)=1] <= x <=[Ceil(1.1)=2] kbps (1 frame/packet) [Ceil(0.12)=1] <= x <= [Ceil(0.7)=1] kbps (10 frames/packet)	Used for RTCP feedback. The full rate is used for 2.5% feedback. The smaller rate is used for feedback every (at least) 5 seconds.
Maximum bit rate for uplink	Equal or higher than guaranteed bit rate	used for RTCP feedback.
Residual BER	10^{-5}	16 bit CRC
SDU error ratio	10^{-4}	
Traffic handling priority	Subscribed traffic handling priority	not relevant
Transfer delay	2 s	

7.4.3 Music only AAC streaming QoS profile

Here we focus on streaming of AAC audio at the bitrate of 52 kbps and a sampling frequency of 24 kHz, which could be suitable for mid-quality stereo music for mobile applications. A frame is composed of 1024 samples and RTP packets contain one single frame. The RTP packetization follows RFC 3016 and each packet is 279 bytes long on average (including payload header and not including RTP/UDP/IPv4 headers). The packet rate is 23.44 packets per second. The total bandwidth for media transmission is 59.9 kbps. About 4.1% bandwith (2.6 kbps) is left for RLC acknowledged mode retransmissions.

Table 4. QoS profile for AAC music streaming at 52 kbps

QoS parameter	Parameter value	comment
Delivery of erroneous SDUs	No	
Delivery order	No	
Traffic class	Streaming	
Maximum SDU size	1400 bytes	
Guaranteed bitrate for downlink	Ceil(61.4)=62 kbps	Including 2.5% for RTCP
Maximum bit rate for downlink	Equal or higher than guaranteed bit rate	
Guaranteed bitrate for uplink	[Ceil(0.12)=1] <= x <= [Ceil(1.5)=2] kbps (1 frame/packet)	Used for RTCP feedback. The full rate is used for 2.5% feedback. The smaller rate is used for feedback every (at least) 5 seconds.
Maximum bit rate for uplink	Equal or higher than guaranteed bit rate	used for RTCP feedback.
Residual BER	10^{-5}	16 bit CRC
SDU error ratio	10^{-4}	
Traffic handling priority	Subscribed traffic handling priority	not relevant
Transfer delay	2 s	

7.4.4 Voice and video streaming QoS profile

The video codec in this case has a bitrate of 44 kbps, with RTP payload packets of 500 bytes (including payload header). The total video bit rate is 47.7 kbps (including RTP/UDP/IPv4 headers). In the same bearer there is an AMR stream at 7.95 kbps with 10 frames encapsulated per RTP packet. The total voice bit rate is 10.1 kbps (including RTP/UDP/IP headers). The total user bit rate is 57.8 kbps. A ~7.3% bearer capacity (4.7 kbps) has been left for RLC Acknowledeged mode retransmissions. The total user bit rate has been computed from the video encoding bit rate,

supposed this is an average bit rate calculated over the sequence length. In case the video encoding bit rate is extracted from the Max_Bitrate in the BitrateBox field of the file format, there might be bearer capacity unused if the difference between such maximum bit rate and the average bit rate of the video stream is large.

Table 5: QoS profile for voice and video streaming at an aggregate bit rate of 57.8 kbps

QoS parameter	Parameter value	comment
Delivery of erroneous SDUs	No	
Delivery order	No	
Traffic class	Streaming	
Maximum SDU size	1400 bytes	
Guaranteed bitrate for downlink	Ceil(59.3)=60 kbps	Including 2.5% for RTCP
Maximum bit rate for downlink	Equal or higher than guaranteed bit rate	
Guaranteed bitrate for uplink	[Ceil(0.12)=1] <= x <= [Ceil(1.5)=2] kbps	Used for RTCP feedback. The full rate is used for 2.5% feedback. The smaller rate is used for feedback every (at least) 5 seconds.
Maximum bit rate for uplink	Equal or higher than guaranteed bit rate	used for RTCP feedback.
Residual BER	10^{-5}	16 bit CRC
SDU error ratio	10^{-4}	
Traffic handling priority	Subscribed traffic handling priority	not relevant
Transfer delay	2 s	

7.4.5 Voice and video streaming QoS profile for GPRS Rel. '97

In this use case it is supposed a 3+1 time slot configuration using coding schemes CS1 and CS2 in GPRS Rel. '97. The peak bit rates are 40.2 kbps for downlink and 13.2 kbps for uplink. The video codec in this case has a bitrate of 30 kbps, with RTP payload packets of 500 bytes (including payload header). The total video bit rate is 32.5 kbps (including RTP/UDP/IP headers). In the same bearer there is an AMR stream at 4.75 kbps with 10 frames encapsulated per RTP packet. The total voice bit rate is 7.3 kbps (including RTP/UDP/IPv4 headers). The total user bit rate is 39.8 kbps. We assume GPRS is configured to use V.42 bis data compression in the SNDCP layer, to allow reduction of the RTP/UDP/IP header size.

Table 6: QoS profile for voice and video streaming at an aggregate bit rate of 39.8 kbps over GPRS Rel. '97

QoS parameter	Parameter value	comment
Service precedence/priority	1	
Delay class	1	
Mean throughput class	17	It means 44 kbps
Peak throughput class	4	It means 64 kbps
Reliability class	3	Unack LLC + Ack RLC modes

7.5 Robust handover management

Handovers are a typical feature of mobile networks, in order to provide mobility to users. Handovers can be perceived as lossless or lossy at the application layer. If they are lossless, the application will experience an increase in the delay/jitter of the packet arrival. Lossy handovers produce breaks in service continuity, which translate in packet losses at the application layer (the amount of losses is equal to the duration of the handover). In particular, inter-system handovers (e.g., between UTRAN and GERAN networks, or between GERAN Rel. '99 and GPRS Rel. '97 networks) can be of long duration (in the order of several seconds).

In order to avoid situations of discontinuous playback, there is the need to smooth out the handover effect from the playback of a streaming session. It must be pointed out that a handover is no different from the link outage effect that a user could experience for example under a tunnel. In this regard, a lossy handover and a period of link outage have the same effect in terms of disruption of playback. A Rel. 5 (or earlier) PSS client with no rate adaptation mechanisms, or

no advanced features for handover management may make use of the available standard methods, in order to handle robustly lossy handovers. In this section an RTSP-based method is described.

A PSS client can detect a lossy handover event by monitoring the buffer level. For example, if the buffer does not receive data for a certain amount X of time (X is an implementation-dependent threshold for the client to understand that the handover event has occurred, and it is required that the client buffer has a size (in time) longer than the handover period), and later it starts to receive data after a certain variable amount Y of time (Y > X, Y is the real duration of the handover period), then the client can trigger an RTSP procedure for robust handover management (the client should verify that the link outage did, in fact, caused loss).

After the handover is over, the PSS client sends a message (resending request) to the PSS server containing the time of the last received media unit before the handover. This information can be delivered using a simple RTSP PAUSE/PLAY messages. An example of such PAUSE and PLAY messages is shown below (last correctly received media unit was at second 28.00):

```
C->S  PAUSE rtsp://example.com/foo RTSP/1.0
      Cseq: 6
      Session: 354832

C->S  PLAY rtsp://example.com/foo RTSP/1.0
      Cseq: 7
      Session: 354832
      Range: npt=28.00-
```

With these messages, the server can re-PLAYs the part of the stream that was lost during handover plus the remainder of the stream. Although a PAUSE message is sent, it is not needed to pause the actual playback in the PSS client, unless the buffer gets empty.

Annex <A>:
Characterisation metrics and testing guidelines

The following set of metrics and testing guidelines are recommended to be used when running PSS characterization future tests.

Guidelines to use case definition:

- Use always PSS Release 5 server.
- For each case first benchmark how a “simple” (implements only mandatory parts of the spec), PSS application would perform.
- The network type and release is specified per each use case
- Specify whether header compression (ROHC) is used/not used

Agreed common settings that should be used to declare a test valid:

- Type of clip to be used (sports, news/weather, movie trailer) – number of scene changes, changing dynamics
- Clip length ~ 2 minutes
- Error concealment is to be used

Issues/Assumptions

- Assess the complexity of the server/client application algorithms that are used in the use cases.
- Assess how much knowledge needs to be there in the application about the bearer implementation options and conditions so that the application can decide to turn the respective critical case handling algorithms/options on, and how feasible it is to get that information.

User perceived streaming quality metrics:

- Number of interruptions in the playout (e.g. rebuffering, long skip of content)
- Playback delay (initial ignalling+buffering time)
- Video frame rate
- Absolute PSNR for video
- PSNR difference between the encoded and the received video (count PSNR for also frames dropped by using the previously received frame)
- Frame error rate for audio

Resource utilisation metrics :

- Amount of data discarded at the receiver
- Under-utilisation?

Information to be included when reporting the test results:

- Diagram for playback, transmission, reception curve (see e.g. Section 7.2)
- Network latency
- Pre-decoder buffer size
- Network buffering assumptions
- Packet loss rate (differentiate losses in the network and packets dropped at the receiver)

- Server characterisation
- Transmission bitrate scheduling model
- VBR or CBR encoding/transmission
- Packetization strategy, packet sizes.

Annex B:
Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
2003-09	SA#21	SP-030443			Presented for approval at SA#21	2.0.0	5.0.0

Editors: Igor Curcio, Viktor Varsa - Nokia Corporation <igor.curcio@nokia.com, viktor.varsa@nokia.com>