3GPP2 C.R1008-0 v2.0

December 2011

cdma2000 Multimedia Services Evaluation Methodology

Revision History

Revision	Description of Changes	Date
Rev 0 v1.0	Initial Publication	January 2007
Rev 0 v2.0	Removal of non-public video test sequences	December 2011

Contents

List of Figures

List of Tables

Foreword

(This foreword is not part of this document.)

This technical report provides common simulation conditions for cdma2000®[1] multimedia services such as Broadcast Multicast Services (BCMCS), Packet Switched Video Telephony/ Multimedia Conversational Services (PSVT/MCS), Multimedia Messaging Services (MMS) [18] and Multimedia Streaming Services (MSS) [19]. The common simulation conditions include source material, pointers to multimedia codecs used in cdma2000 multimedia services, an abstraction of layers below Real Time Transport Protocol (RTP) in wireless Internet Protocol (IP) networks, and suitable objective metrics.

.

[1] cdma2000® is the trademark for the technical nomenclature for certain specifications and standards of the Organizational Partners (OPs) of 3GPP2. Geographically (and as of the date of publication), cdma2000® is a registered trademark of the Telecommunications Industry Association (TIA-USA) in the United States.

1 Introduction

The purpose of this technical report is to document the simulation methodology for cdma2000 multimedia services. The simulations are designed to capture performance characteristics of source codecs, including appropriate RTP packetization schemes used for cdma2000 multimedia services. The layers below the RTP layer are abstracted in a software tool with RTP packets as input, RTP packets as output and appropriate control/configuration parameters to capture packet loss and delay behavior of a given wireless IP network. Appropriate metrics to capture user experience under packet losses and variable delays are also provided. Common source material is proposed for use in the simulations.

1.1 Scope

The scope of this technical report is to provide common simulation conditions to enable repeatable evaluations of source coding and RTP packetization options. It is possible to perform cross verifications. The document includes

Common source material (e.g., video clips)

Pointers to multimedia codecs used in cdma2000 multimedia services, including technical descriptions and reference software when available

An abstraction of layers below RTP in wireless IP networks

Suitable objective metrics

1.2 Requirements Language

“Shall” and “shall not” identify requirements to be followed strictly to conform to this document and from which no deviation is permitted. “Should” and “should not” indicate that one of several possibilities is recommended as particularly suitable, without mentioning or excluding others, that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is discouraged but not prohibited. “May” and “need not” indicate a course of action permissible within the limits of the document. “Can” and “cannot” are used for statements of possibility and capability, whether material, physical or causal.

~~1.2~~1.3 Informative References

The following standards are referenced in this text. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based upon this document are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. American National Standards Institute (ANSI) and TIA maintain registers of currently valid national standards published by them.

[1] ITU-T VCEG-M77, *Common Test Conditions for RTP/IP over 3GPP/3GPP2*.

[2] Parag Agashe, et al, *cdma2000 High Rate Broadcast Packet Data Air Interface Design*, IEEE Communication Magazine, February 2004.

[3] 3GPP2 C.S0024, *cdma2000 High Rate Packet Data Air Interface Specification*.

[4] 3GPP2 C.R1002, cdma2000 Evaluation Methodology. ~~http://3gpp2.org/Public_html/specs/C.R1002-0_v1.0_041221.pdf~~

[5] *Reserved.* ~~3GPP2 C.S0014-A, "Enhanced Variable Rate Codec, Speech Service Option 3 for Wideband Spread Spectrum Digitial Systems"~~

[6] 3GPP2 C.S0014-B v1.0, *Enhanced Variable Rate Codec, Speech Service Option 3 and 68 for Wideband Spread Spectrum Digital Systems*, May 2006.

[7] 3GPP2 C.S0052-A v1.0, *Source-Controlled Variable-Rate Multimode Wideband Speech Codec (VMR-WB), Service Options 62 and 63 for Spread Spectrum Systems*, April 2005.

[8] 3GPP2 C.S0020-A v1.0, *High Rate Speech Service Option 17 for Wide Band Spread Spectrum Communication Systems*, April 2004.

[9] 3GPP2 C.R1009, *cdma2000 Multimedia Services Evaluation Methodology: Software Tools*.

[10] 3GPP2 C.S0054-0 v2.0, *cdma2000 High Rate Broadcast-Multicast Packet Data Air Interface Specification*, August 2005.

[11] 3GPP2 C.S0054-A v1.0, *cdma2000 High Rate Broadcast-Multicast Packet Data Air Interface Specification*, March 2006.

[12] ITU-T P.800, *Methods for subjective determination of transmission quality*.

[13] J.P. Egan, *Articulation Testing Methods*, Laryngoscope, Vol. 58, No. 9, 955-991, 1948.

[14] ITU-T Recommendation H.263, *Video Coding for Low Bitrate Communication*.

[15] ISO/IEC 14496-2:2004, *Information Technology — Generic Coding of Audio-Visual Object — Part 2: Visual*.

[16] ITU-T Recommendation H.264 (2003), *Advanced video coding for generic audiovisual services*~~|~~, or ISO/IEC 14496-10:2003, *Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding*.

[17] ISO/IEC 14496-3:2005, *Information technology - Coding of audio-visual objects - Part 3:Audio*.

[18] 3GPP2 C.S0045-A v1.0, *Multimedia Messaging Service (MMS) Media Format and Codecs for cdma2000 Spread Spectrum Systems*, April 2006.

[19] 3GPP2 C.S0046-0 v1.0, *3G Multimedia Streaming Services*, March 2006.

~~1.3~~1.4 Definitions, Symbols and Abbreviations

This section contains definitions, symbols and abbreviations that are used throughout the document.

~~1.3.1~~1.4.1 Definitions

codec: a system component that encodes and decodes data (usually audio, video, etc.) from one representation to another, often with the goal of saving memory space or transmission bandwidth (compression).

multimedia: a combination of multiple media elements used in a service to enrich the user experience.

~~1.3.2~~1.4.2 Symbols and Abbreviations

3GPP2	Third Generation Partnership Project 2
AAC	Advanced Audio Coding
ARQ	Automatic Repeat Request
AT	Access Terminal
BCMCS	Broadcast-Multicast Services
CIF	Common Intermediate Format
C/I	Carrier to Interference (Ratio)
DCCH	Dedicated Control Channel
DRC	Data Rate Control
EVRC	Enhanced Variable Rate Codec
FER	Frame Error Rate
FL	Forward Link
HE AAC	High Efficiency Advanced Audio Coding
HRPD	High Rate Packet Data
IP	Internet Protocol
ITU	International Telecommunications Union
LSB	Least Significant Bit
MAC	Medium Access Control
MCS	Multimedia Conversational Services
MMS	Multimedia Messaging Service
MPEG	Motion Picture Expert Group
MSB	Most Significant Bit
MSO	Markov Service Option
MSS	Multimedia Streaming Service
MTU	Maximum Transmission Unit
NCIM	Network Client Interface Module
NB	Narrow Band
PCM	Pulse Code Modulation
PDU	Packet Data Unit

pDVD	percentage Degraded Video Duration
PSNR	Peak Signal to Noise Ratio
PSVT	Packet Switched Video Telephony
QCIF	Quarter Common Intermediate Format
QoS	Quality of Service
RL	Reverse Link
RTP	Real-time Transport Protocol
SCH	Supplemental Channel
S/W	Software
TTI	Transmission Time Interval
UDP	User Datagram Protocol
URL	Uniform Resource Locator
VCEG	Video Coding Experts Group
VoIP	Voice over Internet Protocol
VMR	Variable-Rate Multimode
VMR-WB	Variable Rate Multimode Wide Band
WB	Wide Band

2 Source Material

This section ~~contains~~ describes multimedia source materials to be used for test and evaluation purposes for various cdma2000 multimedia services.

2.1 Video

~~This subsection contains v~~Video test sequences can be, but are not limited to, the two described in the following subsections to ~~be used for~~ test and evaluat~~ion purposes for various~~e cdma2000 video applications.~~[2]~~ These two sequences are included in [9].

~~2.1.1 3GPP2 Database Sequence 1: Foreman~~

~~Sequence Name~~	~~foreman~~
~~Spatial Resolution~~	~~QCIF~~
~~Temporal Resolution~~	~~30 fps~~
~~Sequence length~~	~~300 frames; 10 seconds~~
~~Color Space~~	~~4:2:0~~
~~Copyright status~~	~~none~~
~~SHA-1 Hash~~	~~b9cc4accd3c73cec3a14a53d0a0b0a12b5c02c01~~
~~Sequence Characteristics~~	~~amateur, moving head, medium spatial details, camera pan~~
~~Available at (URL)~~	~~3GPP2 internal data base (Dave Singer singer@apple.com)~~
~~Further remarks~~	~~none~~

~~2.1.2 3GPP2 Database Sequence 2: Stunt~~

~~Sequence Name~~	~~stunt_walk_QCIF.yuv~~
~~Spatial Resolution~~	~~QCIF~~
~~Temporal Resolution~~	~~15 fps~~
~~Sequence length~~	~~450 frames; 30 seconds~~
~~Color Space~~	~~4:2:0~~

~~[2] David Singer (singer@apple.com), Apple Computer, provided video sequences, and Thomas Stockhammer (stockhammer@nomor.de), BenQ, provided the checksum numbers.~~

~~Copyright status~~	~~This sequence is provided only for internal 3GPP2 testing purposes only. Other uses, particularly any involving public display, demonstration, or presentation are reserved, and in these cases or if there is any doubt the copyright owners should be contacted for the appropriate permissions.~~
~~SHA-1 Hash~~	~~223f703f886ab40f02c7e4f222c80a7f4d704743~~
~~Sequence Characteristics~~	~~amateur; handheld camera~~
~~Available at (URL)~~	~~3GPP2 internal data base (Dave Singer singer@apple.com)~~
~~Further remarks~~	~~none~~

~~2.1.3 3GPP2 Database Sequence 3: Doctor~~

~~Sequence Name~~	~~doctor_zoom_QCIF.yuv~~
~~Spatial Resolution~~	~~QCIF~~
~~Temporal Resolution~~	~~15 fps~~
~~Sequence length~~	~~450 frames; 30 seconds~~
~~Color Space~~	~~4:2:0~~
~~Copyright status~~	~~This sequence is provided only for internal 3GPP2 testing purposes only. Other uses, particularly any involving public display, demonstration, or presentation are reserved, and in these cases or if there is any doubt the copyright owners should be contacted for the appropriate permissions.~~
~~SHA-1 Hash~~	~~dcb29f3757d8146c69d6e603acaedab7a0e6be91~~
~~Sequence Characteristics~~	~~amateur; handheld camera~~
~~Available at (URL)~~	~~3GPP2 internal data base (Dave Singer singer@apple.com)~~
~~Further remarks~~	~~none~~

~~2.1.4 3GPP2 Database Sequence 4: Walk~~

~~Sequence Name~~	~~walk_friends_QCIF.yuv~~
~~Spatial Resolution~~	~~QCIF~~
~~Temporal Resolution~~	~~30 fps~~
~~Sequence length~~	~~900 frames; 30 seconds~~
~~Color Space~~	~~4:2:0~~

~~Copyright status~~	~~This sequence is provided only for internal 3GPP2 testing purposes only. Other uses, particularly any involving public display, demonstration, or presentation are reserved, and in these cases or if there is any doubt the copyright owners should be contacted for the appropriate permissions.~~
~~SHA-1 Hash~~	~~ee022c5dd68d36bff6860668ffe5b42666412c4c~~
~~Sequence Characteristics~~	~~amateur; handheld camera~~
~~Available at (URL)~~	~~3GPP2 internal data base (Dave Singer singer@apple.com)~~
~~Further remarks~~	~~None~~

~~2.1.5 3GPP2 Database Sequence 5: Crossing~~

~~Sequence Name~~	~~crossing_QCIF.yuv~~
~~Spatial Resolution~~	~~QCIF~~
~~Temporal Resolution~~	~~30 fps~~
~~Sequence length~~	~~600 frames; 20 seconds~~
~~Color Space~~	~~4:2:0~~
~~Copyright status~~	~~This sequence is provided only for internal 3GPP2 testing purposes only. Other uses, particularly any involving public display, demonstration, or presentation are reserved, and in these cases or if there is any doubt the copyright owners should be contacted for the appropriate permissions.~~
~~SHA-1 Hash~~	~~2280746f098c1db92da2318e888dcabbf014df75~~
~~Sequence Characteristics~~	~~semi-professional; static surveillance camera~~
~~Available at (URL)~~	~~3GPP2 internal data base (Dave Singer singer@apple.com)~~
~~Further remarks~~	~~None~~

~~2.1.6~~2.1.1 3GPP2 Database Sequence 1~~6~~: Little Bugs

Sequence Name	bugs_QCIF.yuv
Spatial Resolution	QCIF
Temporal Resolution	25 fps
Sequence length	793 frames; 31.21 seconds
Color Space	4:2:0

~~Copyright status~~	~~This sequence is provided only for internal 3GPP2 testing purposes only. Other uses, particularly any involving public display, demonstration, or presentation are reserved, and in these cases or if there is any doubt the copyright owners should be contacted for the appropriate permissions.~~
SHA-1 Hash	c7135f2a95a563c2ff88c24c1b1a714a580db457
Sequence Characteristics	amateur; handheld camera
~~Available at (URL)~~	~~3GPP2 internal data base (Dave Singer singer@apple.com)~~
~~Further remarks~~	~~None~~

~~2.1.7~~2.1.2 3GPP2 Database Sequence 2~~7~~: Large Bugs

Sequence Name	bugs_CIF.yuv
Spatial Resolution	CIF
Temporal Resolution	25 fps
Sequence length	793 frames; 31.21 seconds
Color Space	4:2:0
~~Copyright status~~	~~This sequence is provided only for internal 3GPP2 testing purposes only. Other uses, particularly any involving public display, demonstration, or presentation are reserved, and in these cases or if there is any doubt the copyright owners should be contacted for the appropriate permissions.~~
SHA-1 Hash	878e6efada8949447c9dfca8dc864f08d4722e1a
Sequence Characteristics	amateur; handheld camera
~~Available at (URL)~~	~~3GPP2 internal data base (Dave Singer singer@apple.com)~~
~~Further remarks~~	~~None~~

~~2.1.8 3GPP2 Database Sequence 8: Little Car~~

~~Sequence Name~~	~~news_car_QCIF.yuv~~
~~Spatial Resolution~~	~~QCIF~~
~~Temporal Resolution~~	~~30 fps~~
~~Sequence length~~	~~916 frames; 30.16 seconds~~
~~Color Space~~	~~4:2:0~~

~~Copyright status~~	~~This sequence is provided only for internal 3GPP2 testing purposes only. Other uses, particularly any involving public display, demonstration, or presentation are reserved, and in these cases or if there is any doubt the copyright owners should be contacted for the appropriate permissions.~~
~~SHA-1 Hash~~	~~2392dc1ce5323d57624500c84ad6aef8ce7963b9~~
~~Sequence Characteristics~~	~~Professional~~
~~Available at (URL)~~	~~3GPP2 internal data base (Dave Singer singer@apple.com)~~
~~Further remarks~~	~~None~~

1 ### ~~2.1.9 3GPP2 Database Sequence 9: Large Car~~

~~Sequence Name~~	~~news_car_CIF.yuv~~
~~Spatial Resolution~~	~~CIF~~
~~Temporal Resolution~~	~~30 fps~~
~~Sequence length~~	~~916 frames; 30.16 seconds~~
~~Color Space~~	~~4:2:0~~
~~Copyright status~~	~~This sequence is provided only for internal 3GPP2 testing purposes only. Other uses, particularly any involving public display, demonstration, or presentation are reserved, and in these cases or if there is any doubt the copyright owners should be contacted for the appropriate permissions.~~
~~SHA-1 Hash~~	~~0d6dfc0ab4745e5ccb0fef0888adebba36c94ff9~~
~~Sequence Characteristics~~	~~Professional~~
~~Available at (URL)~~	~~3GPP2 internal data base (Dave Singer singer@apple.com)~~
~~Further remarks~~	~~None~~

2 ### ~~2.1.10 3GPP2 Database Sequence 10: Africa~~

~~Sequence Name~~	~~BBC_africa_CIF.yuv~~
~~Spatial Resolution~~	~~352x198~~
~~Temporal Resolution~~	~~24 fps~~
~~Sequence length~~	~~771 frames; 32.03 seconds~~
~~Color Space~~	~~4:2:0~~

~~Copyright status~~	~~This sequence is provided only for internal 3GPP2 testing purposes only. Other uses, particularly any involving public display, demonstration, or presentation are reserved, and in these cases or if there is any doubt the copyright owners should be contacted for the appropriate permissions.~~
~~SHA-1 Hash~~	~~f1650fc6cae1f476ff15af7eed6c697297bd0090~~
~~Sequence Characteristics~~	~~professional~~
~~Available at (URL)~~	~~3GPP2 internal data base (Dave Singer singer@apple.com)~~
~~Further remarks~~	~~The handling of this spatial resolution for reference encoders and decoders has to be clarified.~~

2.2 Speech

This document does not specify any speech source material for generic evaluation purposes. It is a common practice to identify source material for a specific test on an as-needed basis. An example of a speech source to be used in subjective evaluation is given in [12]. The Harvard database [13] of phonetically balanced sentences is an example of the speech source described in [12] and is also commonly used in subjective evaluations.

2.3 Audio

This document does not specify any audio source material for generic evaluation purposes. It is common practice to identify source material for a specific test on an as-needed basis.

3 Codec

This section describes the multimedia codecs used in cdma2000 multimedia services such as BCMCS, PSVT/MCS, MMS [18] and MSS [19].

3.1 General Description

This section describes reference implementations of codecs used in cdma2000 multimedia services.

3.2 Speech

Speech codecs used in cdma2000 multimedia services are described in EVRC [6]~~[5]~~, EVRC-B [6], VMR-WB [7], and 13k [8].

3.3 Video

Video codecs used in cdma2000 multimedia services are described in H.263 baseline [14], H.263+ [14], MPEG-4 SPL0 [15], and H.264 baseline [16].

3.4 Audio

Audio codecs used in cdma2000 multimedia services are described in AAC [17] and HE AAC [17].

This page intentionally left blank.

4 Channel

This section describes channel models used for the proposed evaluation methodology.

4.1 Dedicated Channel

This section presents simulation assumptions for generating error rate for the forward DCCH and supplemental channels of cdma2000. Error masks for typical conditions for use in video characterization are included.

4.1.1 Description of Traces

Error masks that can be used for video simulations on dedicated channel assignments (SCH and DCCH) at various bit rates are provided in [9]. When a video packet occupies both SCH and DCCH, the application packet will be lost if there is an error either on SCH or on DCCH. But the losses on SCH and DCCH will be uncorrelated. In order to simulate this, it is propose to use time reversed error patterns provided in *.rev files in [9].

FERp: p = {1, 3 5 and 7} are the outer loop target errors.

CMq: q = {0, 1, 2 3, 4} corresponding to channel models A, B, C, D and E

gm6: Geometry at 6 dB

AS1: Active set =1; No soft handoff.

***.out**: The error masks as explained above.

***.rev**: Error masks in reverse direction, to generate uncorrelated errors

4.1.2 Simulation Methodology

Frame decoding error events are generated in a system-level simulation. A system-level simulation is run, and the decoding successes of each 20ms frame are recorded in the form of ‘0’ and ‘1’ for each 20ms, thereby producing an “error mask”. The error mask is then fed into the video simulation to model air interface errors. In those simulations, all the bits carried in an application layer packet containing the 20ms frame are discarded when the error mask indicated that the frame is in error. This is typically one RTP/UDP/IP packet containing the 20ms frame that was in error. Although the error mask is only generated for one rate, it is applied to any rate modeled.

The error masks are generated by storing the errors seen by one 9.6 kbit/s RC3 user placed in a 19-cell network with wrap-around. All cells are fully loaded with background 1xEV-DV data traffic. T_ADD= -18 dB is used for all members of the active set. A Rician factor of K=5 (in linear) is used for channel model A. All other parameters are according to the Evaluation Methodology [4] developed by 3GPP2 TSG-C WG3, and the link error curves are also provided by 3GPP2 TSG-C WG3.

Error masks are generated and used on video sequences for users with the following parameters:

Channel models: A (with Rician K=5), B, C, D, and E.

The Evaluation Methodology describes these channel models in detail. The next section gives a brief overview of these channels.

Geometry: 6 dB.

The geometry is the ratio of the average total received power from the sectors in the active-set to the average of all other received power. The geometry is therefore some measure of the location of the user, in term of C/I.

This corresponds to a relatively good geometry for a user in one-way or multiple-way soft-handoff.

Number of soft-handoff legs: 1

Channel models with multiple paths already approximate soft-handoff. For simplicity it is therefore recommended to not use soft-handoff cases.

Outer-loop target FER: 1%, 3%, 5% and 7%

This parameter depends on the operation point that is desired. Typically the fundamental channels are operated at 1% and the supplemental channels are 5% FER.

Following is a short description of the 5 channel models.

Table 1 Propagation Channel Models

Channel Model	Description	Multipath Profile	Speed (km/h)	Fading Type
Model A	Pedestrian-like, single path with losses	M0	3	Rician, K=5
Model B	Slowly moving vehicle, many multipaths	M1	10	Rayleigh
Model C	Medium speed vehicle, some multipath	M2	30	Rayleigh
Model D	Rapidly moving vehicle, single path with losses	M0	120	Rayleigh
Model E	User at a fixed location, single path	M3	Doppler = 1.5 Hz	Rician, K=10

Table 2 Multipath Profiles

Multipath Profile				
M0	-0.06			-18.8606
M1	-1.64	-7.8	-11.7	-10.9151
M2	-0.9	-10.3		-10.2759
M3	0			

Table 1 provides a brief description of the five suggested propagation channel models. Each model is associated with a multipath profile M0 through M3 from Table 2, a velocity which determines the rate at which the paths should fade, and a fading type which describes the statistics of the fading.

The multipath profiles of Table 2 describe the average power captured by each of up to 3 possible fingers, and the fraction of un-recovered power. For example, multipath profile M2 consists of one finger capturing on average -0.9 dB of the total received power, one finger capturing on average -10.3 dB of the total received power, then -10.2759 dB of the total received power is un-captured by any finger and therefore acts as interference.

For a given channel model, the energy captured by the various fingers as well as the FURP, each fade independently and according to the velocity and fading type specified.

4.2 HRPD Shared Channels

This section presents simulation models for determining the timing modifications and error events on RTP packets for the forward-link (FL) and the reverse-link (RL) of cdma2000 HRPD [2], [3] in order to simulate multimedia services over such a system. While the models in this section can be applied to any multimedia services, a service (e.g., VoIP) which has small RTP packet sizes may use a simplified methodology described in section 5.3.4. The models in this section can be used to simulate RTP timing modifications and losses in order to simulate delivery of cdma2000 multimedia services.

Timing modifications in a shared channel with packet-based transport introduce variations in end-user packet receiver timing. This is known as delay jitter. For real-time multimedia services, a large delay jitter affects both the service quality and the conversational quality. The sources of jitter to an RTP packet in HRPD are:

The delay before the packet can be scheduled to be transmitted on the shared channel – i.e., scheduling delay – and may be introduced by either the scheduler on the FL or the interlace structure on both the FL and RL,

The delay in transmitting the packet over the shared channel and successfully receiving it, i.e., transmission delay (several factors affect the transmission delay in HRPD system, e.g., channel condition, Hybrid-ARQ early termination, current resource allocation in the RL, and retransmission mechanism such as Delayed-ARQ).

RTP packet losses in an HRPD system can be categorized into the following types:

Packet losses due to the wireless channel, which occur because while in HRPD the power control loops will try to adjust the physical layer packet losses to a certain target and consequently the actual packet losses may depend on the physical layer packet sizes chosen and the actual wireless channel (another important factor impacting RTP packet loss rate is the means by which the RTP packets are fragmented across different physical layer packets);

Packet losses due to excessive scheduling delay – packets that take too long to be scheduled on both the FL and the RL may be dropped when they exceed a pre-defined delay threshold (this is done in order to avoid wasting bandwidth by transmitting stale packets that are no longer useful).

Since there are many factors that can affect multimedia services quality, accurate models of the FL and RL which can accurately determine both the timing modification and packet losses in an HRPD system are needed.

4.2.1 Reverse-link Model

In this section, a model of the RL of an AT using Subtype 3 Reverse Traffic Channel MAC Protocol (RTCMAC) specified in [3] is considered. RTCMAC Subtype 3 provides a comprehensive mechanism to support QoS for delay-sensitive services on the RL. The model of RTCMAC Subtype 3 incorporates the following salient features that may affect multimedia service quality.

A transmitted MAC packet size is realistically chosen from a set of available MAC packet sizes in each slot. In a real system, the selected MAC packet size at any given moment depends on the available application payload and MAC allocation. This effect is more pronounced for services which might have a large dynamic range of RTP payload sizes such as video telephony.

Both the scheduling delay and the transmission delay are determined from the interlace structure, termination statistics, and selected MAC packet sizes. Combining the termination statistics with the appropriately chosen transmitted MAC packet sizes will also yield realistic RTP packet loss events.

The MAC allocation is determined using a token bucket similar to RTCMAC Subtype 3. This allows the allocated resource in a shared channel to adapt to bursty sources accordingly, while still regulating the long-term resource allocation. This is a unique feature to packet-switched, wireless systems such as HRPD.

The model assumes that the RL flow under consideration will not react to the Reverse Activity Bit [3] from the AN, i.e., the RTCMAC does not react to the congestion in the RL. The variability in the channel is assumed to be smoothed out by the RL power-control, i.e., the AT can transmit at any packet sizes allowed by the RTCMAC regardless of its geometry as long as the AT still has enough power-headroom.

The model requires only a few input parameters, i.e., the token bucket parameters and termination statistics. The termination statistics are pre-computed through a detailed physical-layer simulation. This allows the model to be realistic with regards to RTP packets behavior that affects multimedia services without being complicated.

4.2.1.1 RL Model: RL Interlace Structure

The reverse-link MAC layer of HRPD consists of three interlaces where each interlace occupies a subframe, i.e., four physical-layer slots or 6.67 ms. A reverse-link MAC packet is composed of four subpackets (S-pkt), where each subpacket occupies a subframe. An RTP packet may be fragmented into several MAC packets as shown in Figure 1 where the payload size of each MAC packet is determined from the token bucket mechanism described in Section 4.2.1.2. Each subpacket is independently decodable to allow early termination of the reverse-link MAC packet to take advantage of variation in the reverse-link channel. Early terminated packet frees up the interlace and allows new packets to be transmitted on the reverse-link. Early termination will be described in Section 4.2.1.3.

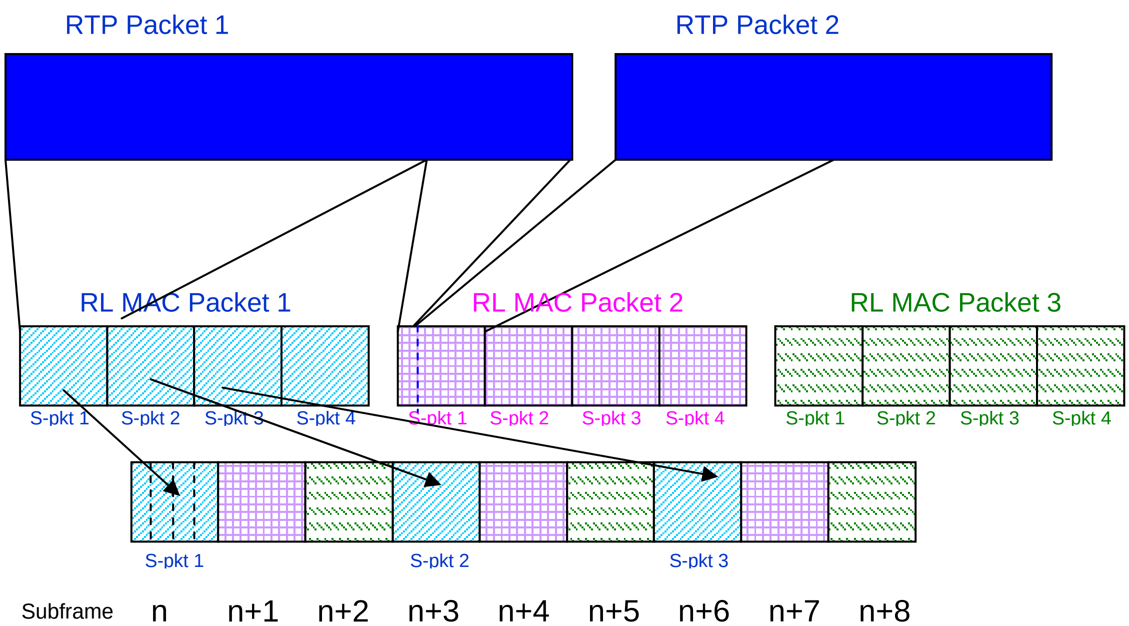

Figure 1 Interlace Structure and RTP Packet Fragmentation in the HRPD Reverse-link Model

Note: the dashed vertical lines shown above in S-pkt 1 represent slots and are not shown in subsequent sub-packets for clarity.

4.2.1.2 RL Model: RL MAC Packet Size Determination

The RL MAC packet size is determined using a token bucket mechanism which is simplified from the specification in [3]. The number of tokens in the bucket available in subframe n, $T_{avail}(n)$, determines the largest RL MAC packet size that can be transmitted in a particular subframe, i.e., the RL MAC needs to withdraw T_k tokens required to transmit a subpacket of packet index k where k = 1,…,12 is the packet index listed in Table 3. The values T_k, k = 1,…,12 are parameters which are set based on the desired packet termination probabilities. The default values of T_k, k = 1,…,12 are given in [3].

Let T_{outmax} denote the largest number of tokens that can be withdrawn from the bucket for a given subframe. If a subframe n is free for a new MAC packet allocation then the RL MAC chooses T(n) tokens to be withdrawn from subframe n for transmission of a MAC subpacket, such that for the 1st subpacket

$$T(n) = T_k$$

for some k, where T_k is no greater than $\min[T_{avail}(n)/4 + T_{in}, T_{outmax}]$. The RL MAC chooses k to be the largest MAC packet index such that no MAC packet with a smaller index can transmit all the available data. The amount of payload transmitted in this MAC packet is then determined from Table 3.

If the RL MAC transmits nothing in the subframe, then sets T(n) = 0. If the subframe contains the 2nd, 3rd, or 4th subpacket, then T(n) should~~all~~ be the same as the 1st subpacket for that MAC packet. At beginning of the subframe n+1, number of the tokens in the bucket is updated using the following formula:

$$T_{avail}(n+1) = \min[T_{avail}(n) + T_{in} - T(n), T_{max}],$$

where T_{in} is the count of tokens added to the bucket (which will regulate the long-term resource allocation to the AT) , and T_{max} is the bucket size. The conceptual diagram of the token bucket for the RL MAC model is shown in Figure 2.

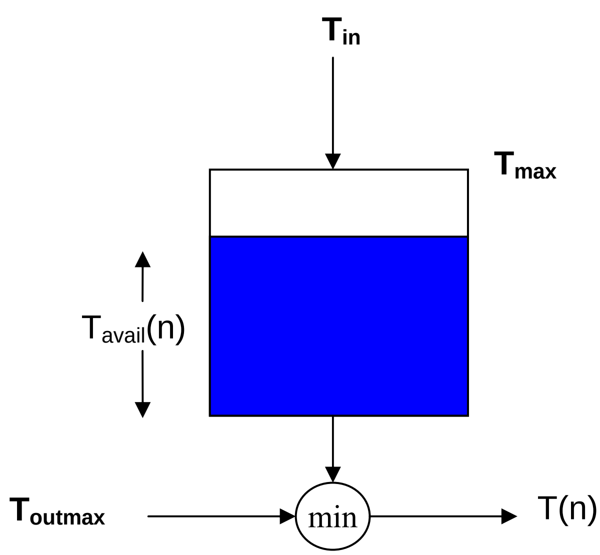

Figure 2 Token Bucket Mechanism in the HRPD Reverse-link Model

Table 3 Reverse-Link Packet Index and Application Payload Size

Packet Index (k)	Physical Layer Packet Size (bits)	Application Payload Size (bits)
1	128	64
2	256	192
3	512	448

Packet Index (k)	Physical Layer Packet Size (bits)	Application Payload Size (bits)
4	768	704
5	1024	960
6	1536	1472
7	2048	1984
8	3072	3008
9	4096	4032
10	6144	6080
11	8192	8128
12	12288	12224

4.2.1.3 RL Model: Timing Modification and Packet Loss Determination

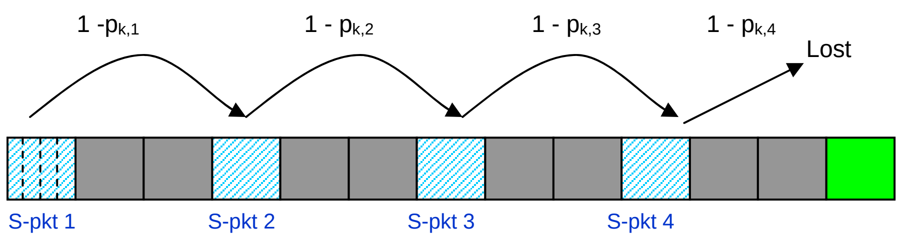

Figure 3 Probabilistic Model for Early Termination Determination

In the RL MAC each MAC packet may successfully terminate before all four subpackets are transmitted. A successfully terminated MAC packet frees up the interlace and allows new MAC packets to be transmitted. This is modeled by using the probabilistic model as illustrated in Figure 3. The i^{th} subpacket, $i = 1,\ldots,4$, may successfully terminate with probability $p_{k,i}$ where $k = 1,\ldots,12$ is the packet index. RL MAC packet that does not terminate successfully after the 4^{th} subpacket is lost. The probability $p_{k,i}$, $k = 1,\ldots,12$, $i = 1,\ldots,4$, are input parameters to the model. They depend on the parameters T_k, $k = 1,\ldots,12$ and are determined from a detailed link-level simulation.

The timing modification for an RTP packet delay is the maximum delay over all MAC packets which include data from the RTP packet. An RTP packet error occurs if there exists a lost MAC packet which contains data from the RTP packet.

4.2.2 Forward-link Model

In this section, a model of the FL of an AT using Enhanced Forward Traffic Channel MAC Protocol (FTCMAC) specified in [3] is examined. The model is based on the results generated from a detailed network-level simulation of an HRPD system. The model can be separated into two distinct parts; namely, the scheduling delay/loss and the transmission delay/loss. The overall timing modification of an RTP packet on the FL is the sum of the scheduling delay and transmission delay.

In general, a model which can capture a realistic scheduling delay is very complex. However, note that in a typical scheduler for delay-sensitive traffic, scheduling opportunities (i.e., the chance for a flow to be selected for transmission in a given slot) for delay-sensitive flows only take into account the head-of-queue delay and do not change with packet sizes. Therefore, the scheduling opportunities can be sampled by sending a small packet at regular intervals in the detailed network-level simulator to find out how long the scheduling delay would be if a packet was to arrive at that time with the given load and channel condition. The recorded scheduling opportunities can then be used later for RTP timing modification, as it can be assumed that any arriving RTP packets around a scheduling opportunity will experience the recorded scheduling delay associated with the scheduling opportunity. Any RTP packets that have scheduling delay larger than the delay threshold are discarded.

Modeling the transmission delay, on the other hand, is straightforward and is specified in detail in the HRPD specification [3]. Unlike the scheduling delay, the transmission delay of each RTP packet depends on the size of the RTP packet, the channel condition, and the interlace structure. The application payload size and the nominal transmit duration of each FL packet are determined based on the value of the Data Rate Control (DRC) index in the slot. The DRC index in each slot and the corresponding Hybrid-ARQ early termination result can be recorded from a detailed link-level simulator and depends on the physical channel model and the geometry of the AT. By combining the DRC information and packet early termination result with the interlace structure, accurate determination of the packet fragmentation and the overall transmission delay of each RTP packet can be.achieved.

4.2.2.1 FL Model: FL Interlace Structure and Packet Size Determination

The forward-link MAC of HRPD consists of four interlaces where each interlace occupies a physical-layer slots or 1.67 ms. A forward-link MAC packet may have different nominal transmit duration, i.e., span. The span for each forward-link MAC packet is determined from the DRC index, i.e., the indicator of the forward-link quality to the AT, as shown in Table 4. An RTP packet may be fragmented into several MAC packets as shown in Figure 4 where the payload size of each MAC packet is determined from the DRC index in Table 4. After a fragment has been scheduled, the following fragments may be delayed for additional C slots[3] before being scheduled Similar to the reverse-link, each forward-link MAC packet may early terminate successfully before the MAC packet is transmitted on all slots for its span. An early terminated packet frees up the interlace and allows new packets to be transmitted. Early termination results are determined from a detailed link-level simulation and are parts of the input parameters to the model.

[3] C is a predetermined constant value for the selected loading conditions that characterizes how subsequent fragments of an application packet are delayed by higher priority traffic.

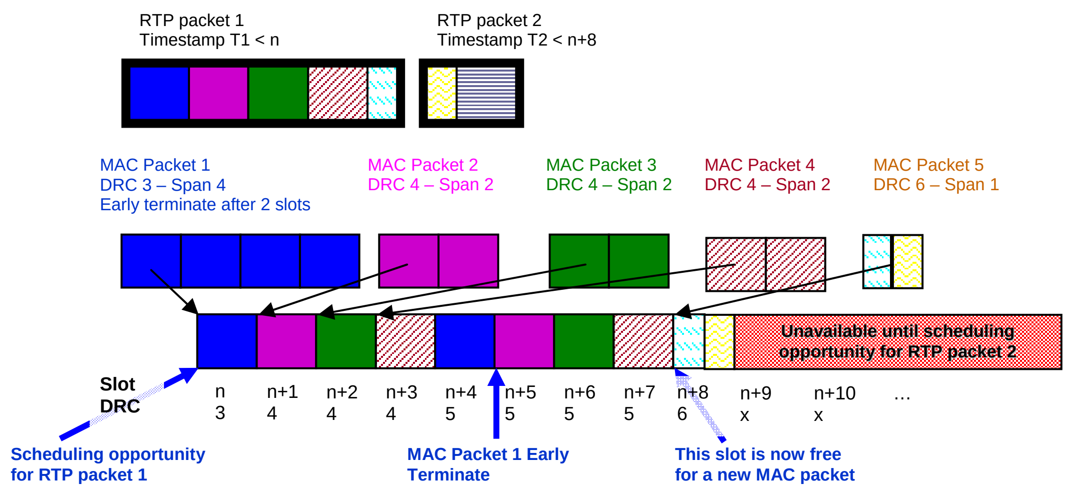

Figure 4 Interlace Structure and RTP Packet Fragmentation in the HRPD Forward-Link Model

Table 4 DRC Index and the Corresponding Span and Application Payload Size

DRC Index	Span (slots)	Application Payload Size (bits)
0	16	952
1	16	952
2	8	952
3	4	952
4	2	952
5	4	1976
6	1	952
7	2	1976
8	2	3000
9	1	1976
10	2	4024
11	1	3000
12	1	4024
13	2	5048
14	1	5048

4.2.2.2 FL model: Timing Modification and Packet Loss Determination

For an arriving RTP packet, the packet is first delayed according to the pre-determined scheduling delay of that arriving slot which is part of the input parameters to the model. After the RTP packet is delayed accordingly, it is scheduled for transmission on the next available interlace. If there are many RTP packets available to be transmitted in the available interlace, the earliest arriving packet is transmitted first. The RTP packet may be fragmented to several forward-link MAC packets where the size of each MAC packet is determined from the DRC index of the first slot of the transmission. The timing modification for an RTP packet delay is the scheduling delay plus the maximum transmission delay over all MAC packets which include data from the packet as shown in Figure 5.

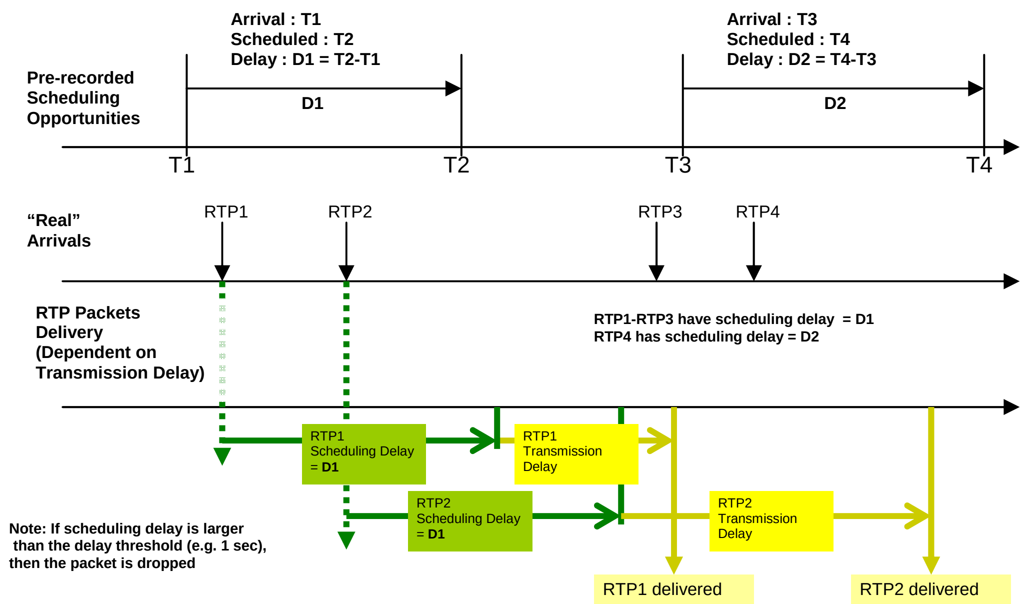

Figure 5 Scheduling Delay Determination in the Forward-Link Model

An RTP packet error occurs if any MAC packet that contains data from the RTP packet is lost. A forward-link MAC packet is considered lost when its Delayed-ARQ (D-ARQ) retransmission is lost. D-ARQ improves PER without significantly incurring additional delay. The payload of a forward-link MAC packet that is transmitted the first time and lost is immediately queued in the D-ARQ queue for retransmission. Data in D-ARQ queue is transmitted before any first-time transmission data and can be transmitted on the next available interlace without scheduling delay. The payload and span of the retransmitted packet, however, is still determined from the DRC index of the slot for the retransmission. A lost D-ARQ packet may not be retransmitted.

4.2.3 End-to-end Channel Model for VoIP

The channel model for VoIP can be appropriately represented by using delay profiles, which capture the delay as well as error behavior of RTP packets under various conditions of channel quality and network loading.

This section describes details of input data to be used to measure the performance of VoIP terminals. It describes different cases that the input data covers (i.e., cases of network loading, channel conditions etc).

Note: these input data are not exhaustive; additional data may be added when available.

4.2.3.1 Delay Traces for HRPD rev A

The input delay traces cover a combination of cases involving different levels of network loading and different channel qualities. In addition, different scenarios such as mobile-to-mobile, mobile-to-land calls are considered. These delay traces have been obtained by simulations run according to the simulation framework, defined in [4].

In mobile-to-land calls, the land part of the call is assumed to be equivalent to just a fixed delay. This is a fair assumption since even if the land part of the call has jitter, this is likely to be much less than that introduced by the mobile part of the call. Thus, the overall jitter will be dominated by the mobile part of the call. Note that the main advantage of including mobile-to-land delay profiles is to make this document also useful for media gateways, which will need to accommodate the jitter arising only from the reverse link.

One user in each sector (57 sectors) is simulated using full-rate frames, while all other users are modeled using the Markov Service Option (MSO) source, as defined in the simulation framework [4]. Using only full-rate frames in the simulation enables the delay profiles to be mapped on to any speech recording. Thus, the delay profiles are generated from full-rate frames users. The simulation conditions used to generate the delay profiles as well as the FER and delay characteristics of the delay profiles (generated from the full-rate frames user) are shown below. Also shown with each delay profile is the FER percentile of the delay profile among all simulated users for that loading and scenario (Mobile-to-land or Mobile-to-mobile).

Mobile-to-mobile, Half Loading[4], VoIP Traffic Only, FER = 0.047%, Delay Std. Dev = 14.60 msec, FER is in 5th percentile[5].

Mobile-to-mobile, Half Loading, VoIP Traffic Only, FER = 0.192%, Delay Std. Dev = 15.04 msec, FER is in 50th percentile.

Mobile-to-mobile, Half Loading, VoIP Traffic Only, FER = 0.287%, Delay Std. Dev = 16.92 msec, FER is in 85th percentile.

Mobile-to-mobile, Half Loading, VoIP Traffic Only, FER = 0.503%, Delay Std. Dev = 19.95 msec, FER is in 95th percentile.

Mobile-to-mobile, Half Loading, VoIP Traffic Only, FER = 2.327%, Delay Std. Dev = 24.41 msec, FER is in 99th percentile.

Mobile-to-mobile, Full Loading[4], VoIP Traffic Only, FER = 0.023%, Delay Std. Dev = 16.92 msec, FER is in 5th percentile.

[4] Full sector loading is 44 VoIP calls for HRPD RevA. Half sector loading is 22 VoIP calls.

[5] x% percentile means that x% users have better FER than the selected user. The number of samples for the percentile is # of users x # of sectors.

Mobile-to-mobile, Full Loading, VoIP Traffic Only, FER = 0.216%, Delay Std. Dev = 20.92 msec, FER is in 50th percentile.

Mobile-to-mobile, Full Loading, VoIP Traffic Only, FER = 0.384%, Delay Std. Dev = 19.25 msec, FER is in 85th percentile.

Mobile-to-mobile, Full Loading, VoIP Traffic Only, FER = 1.151%, Delay Std. Dev = 25.72 msec, FER is in 95th percentile.

Mobile-to-mobile, Full Loading, VoIP Traffic Only, FER = 3.214%, Delay Std. Dev = 29.16 msec, FER is in 99th percentile.

Mobile-to-land, Half Loading, VoIP Traffic Only, FER = 0.019%, Delay Std. Dev = 14.07 msec, FER is in 5th percentile.

Mobile-to-land, Half Loading, VoIP Traffic Only, FER = 0.154%, Delay Std. Dev = 14.07 msec, FER is in 50th percentile.

Mobile-to-land, Half Loading, VoIP Traffic Only, FER = 0.271%, Delay Std. Dev = 14.61 msec, FER is in 99th percentile.

Mobile-to-land, Full Loading, VoIP Traffic Only, FER = 0.038%, Delay Std. Dev = 14.42 msec, FER is in 5th percentile.

Mobile-to-land, Full Loading, VoIP Traffic Only, FER = 0.174%, Delay Std. Dev = 15.03 msec, FER is in 50th percentile.

Mobile-to-land, Full Loading, VoIP Traffic Only, FER = 0.367%, Delay Std. Dev = 14.77 msec, FER is in 99th percentile.

4.2.3.2 Input Data Format

Each input delay profile consists of packet arrival information as well as associated speech data. The delay profile files have the following format:

Table 5 Format of Input Files

RTP Sequence Number	RTP Timestamp	Arrival Time (msec)
20	1000	202.0
23	1480	224.1
22	1320	224.8
24	1640	245.4
25	1800	245.4

Note that in the delay profile file, packet information is shown in increasing order of time of arrival of packets. Also, some packets may be missing from the input file as they may be erased on the physical layer (e.g., packet 21 in Table 5). Packets may also arrive out of order (e.g., packets 23 and 22 in Table 5).

4.3 Broadcast Channel

This section presents a simulation model to evaluate the performance of multimedia services over HRPD Broadcast-Multicast Services (BCMCS) [10] and Enhanced Broadcast-Multicast Services [11]. Traces that can be used for simulating typical BCMCS channel behavior at various bit rates are provided in [9].

4.3.1 Simulation Procedure

4.3.1.1 Basic Setup

The system layout consists of 19 hexagonal cells, each with 3 sectors, and with wrap-around. The central cell is surrounded by 6 cells of the first tier and 12 cells of the second tier.

A total of M (e.g., 1000) mobile stations are uniformly dropped into this system.

Since there is no interaction between mobile stations in the BCMCS system simulation, mobile stations may be simulated in the system either concurrently in one simulation, or individually in multiple simulation runs.

The power and delay of the multipath components from all the sectors are generated for each mobile station. For basic BCMCS, the combined rake signal-to-interference-noise ratio (SINR) for all the sectors in the active set is computed. For EBCMCS, the combined energy at a mobile station from all the 57 sectors is computed. The mobile stations are then sorted in ascending order of this energy. In order to reduce simulation time, N (e.g., 100) mobile stations with the lowest power may be selected.

4.3.1.2 Generating Error Masks

Recall that the BCMCS physical layer consists of four interlaces, where each interlace occupies one physical layer slot of 1.666…ms. Each slot is further divided into two half-slots.

The simulation period is S slots, leading to S/4 slots per interlace. The channel models defined in Table 1 are assumed from each sector to each user. A physical layer simulation is used to compute the SINR for each user for each slot. This computation is performed for relevant channel models.

An error mask is generated from the SINR trace by selecting a particular bitrate and a SINR threshold. The bitrate determines the number of slots required to transmit a physical layer packet. The SINR threshold determines the probability with which the packet will be correctly decoded at the mobile station. The SINR threshold value is selected based on link-level curves, Doppler and model-induced penalties.

For each packet, the error mask contains a '0' if the accumulated energy of all the slots of the packet exceeds the SINR threshold. It contains a '1' if this accumulated energy falls below the SINR threshold.

The total number of packets per interlace in the simulation is:

P = Total number of slots / Total number of interlaces / Slots per packet

4.3.1.3 Mapping Application Layer Packets to Error Traces with Reed-Solomon Outer Code

Application-layer packets are framed according to the Broadcast Framing Protocol. These packets are then encoded using an (N, K, R) Reed-Solomon code. An Error Control Block (ECB) is created as described in [10]. The number of MAC packets in an ECB row is determined by the parameter MACPacketsPerECBRow. The rows of the ECB form the payload for physical layer packets. The number of physical layer slots required to transmit a MAC packet is determined by the transmission rate.

To find application-layer packet errors, the error trace for the bitrate being used is looked up. If the error trace entry for a physical layer packet is '1', then the corresponding MAC packet and Reed-Solomon code word are treated as lost. If more than R Reed-Solomon codewords are lost, then the corresponding application-layer packet is treated as lost.

4.3.1.4 Output Error Traces

The resulting physical layer error traces corresponding to different channel models are available in [9]. These can be used with the software tools in [9] to model application packet errors for cdma2000 broadcast channels.

5 Simulator

This section describes network simulator to be used for evaluation of multimedia services.

5.1 Network Simulator

Network simulator is used to simulate RTP losses and timing modifications to simulate delivery of cdma2000 multimedia services.

Sections 5.3.1-5.3.3 describe a method to introduce RTP packet losses in bitstreams to simulate delivery of multimedia services on the dedicated channel of 3GPP2 networks. It is based on the VCEG network simulator [1], adopted by VCEG during the development of H.264/AVC codec. This simulator primarily addresses packet losses for video streams over RTP/UDP/IP transport. When a physical layer packet is lost over the air interface, it results in the loss of one or more complete RTP packets that contain portions of the physical layer packet. Here, a physical layer packet is called Packet Data Unit (PDU). This approach can be used for other data also when delivered over RTP/UDP/IP transport. The error insertion device in [1] has been modified to: (i) enable use of packet error masks, (ii) enable the simulator to perform correctly when lower layers support multiple packet sizes for a given transmission time interval (TTI) on the physical layer, and (iii) to use decoders that do not have error detection capability by providing a "forbidden bit" to indicate packet losses. Details on modifications are described in the following.

Enhancements to the RTP network simulator are proposed to enable it to also support shared channels. In dedicated channels, packets are delivered periodically, once every TTI. The only impairment introduced by the channel is dropped packets. However, in shared channels, an additional impairment of non-periodic packet arrival (or packet jitter) is also introduced in addition to dropped packets. The timestamp of the network simulator packet is modified to incorporate the varying packet delay introduced by the wireless system to support shared channels. This timing modification is described in 5.3.4.

5.2 S/W Usage Description

Software usage description and command line options are described in readme.txt included in [9].

5.3 Simulator Description

5.3.1 Frame Dropping Modification

The original simulator required bit error masks, i.e., the sequence of '0' and '1' indicating if a bit was in error or not. The error mask files were also required to be in binary format. An additional feature was added to the simulator enabling it to read ASCII error masks. Its operation was also modified so that the ASCII mask values were interpreted as corresponding to packet losses. Hence, the modified simulator only needs to read one mask for every PDU. If the mask is '1", the PDU is dropped else it is not. When a PDU is in error, the RTP packet(s) corresponding to it is (are) dropped (i.e., not written to output file). This operation is shown in Figure 6. As before if the end of file is reached and there are more PDUs to be transmitted, the simulator continues reading from the beginning of the error mask file.

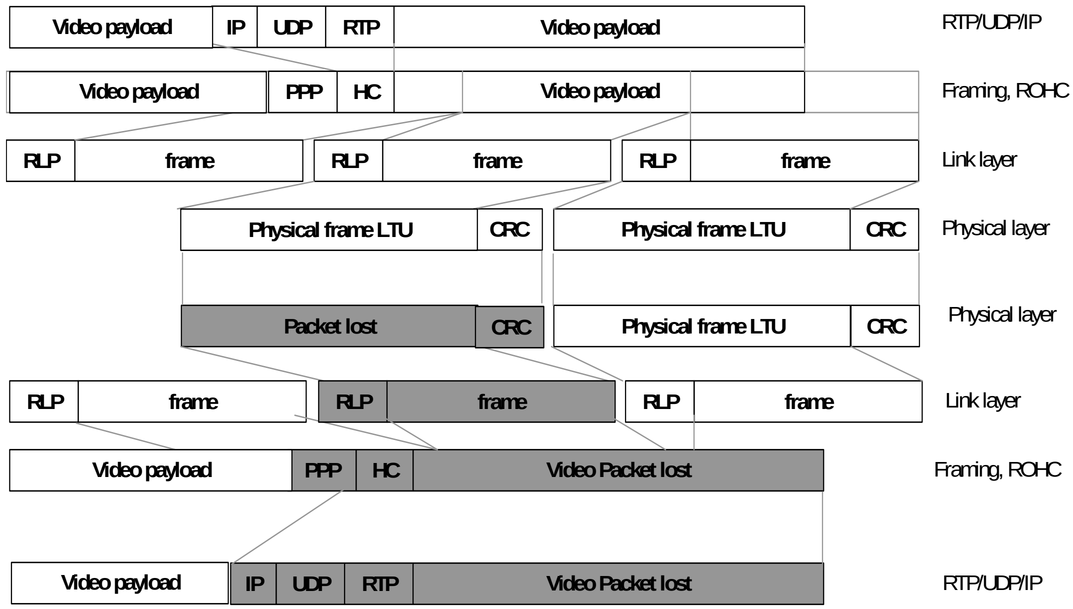

Figure 6 Network Simulator Operation.

Note: The dropped portions of the bitstream at different layers are shown shaded.

5.3.2 Support for Multiple PDU Sizes

For the original simulator, only one PDU size needed to be specified. All RTP packets were broken up into these equal-sized PDUs. In the EBR mode of operation, a video encoder generates RTP packets corresponding to one of several fixed available PDU sizes. Hence a RTP packet is completely contained in a single PDU. To enable this mode of operation all possible PDU sizes need to be specified

When multiple PDU sizes are specified, each RTP packet is expected to be transmitted in a single PDU. One mask is read for each RTP packet and the smallest PDU that can contain the entire RTP packet is considered to be lost over the physical layer.

In the EBR mode of the simulator, the PDUs themselves are assumed to provide framing information. Hence PPP is not required for EBR. As a result, the packet size expansion feature due to PPP has been turned off in the network simulator.

5.3.3 Forbidden Bit Mode for Error Detection

Typically, reference decoder implementations from ITU and MPEG do not handle non-compliant bitstreams, i.e. bitstreams with packet losses. Yet, a common platform is desirable to assess the quality in the presence of typical cdma2000 channel errors. In order to facilitate this, another mode was added to the original VCEG software simulator. In this mode, the RTP packets are not removed from the bitstream, but a “forbidden bit” is used to indicate to the decoder that the RTP packet contains the data that would have been lost over the air. The decoder can simply discard the decoded slices and copy co-located macroblocks from the previously reconstructed YUV frames. The modifications for forbidden bit are as follows:

Only the 16 LSBs of the size field are used to indicate packet size in octets. This is adequate as RTP packets can not exceed the MTU size. The 16 MSB bits are set to 0x0000 for non-erroneous packets and are set to 0x0001 for erroneous packets. The forbidden bit setting (on or off) is controllable from the settings file.

5.3.4 Timing Modification for Multimedia

This section describes the testing methodology for evaluating multimedia clients operating over HRPD Rev A. The block diagram of the testing methodology is shown in Figure 7. This diagram depicts a logical representation of the test setup for multimedia clients. Note that all the inputs and outputs for the simulation are in network byte order (independent of platforms used).

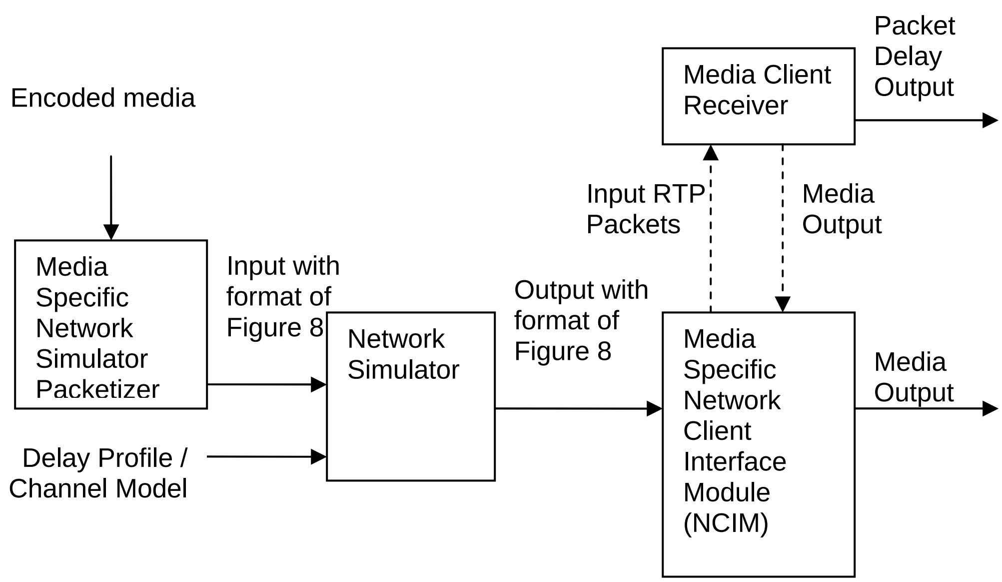

Figure 7 Test Setup for Multimedia Clients

The above set-up can be defined for any media. The test set-up is defined for VoIP in sections 5.3.4.1 - 5.3.4.4.

5.3.4.1 Network Simulator Packetizer for VoIP

For VoIP simulation methodology, the first block in Figure 7 is the media specific Network Simulator Packetizer. The Network Simulator packetizer packetizes the input speech file in the format of Figure 8. The inputs and outputs of the Network Simulator Packetizer are:

Input File:

Encoded speech file: in the format of . The codecs used in cdma2000 multimedia services (13K [8], EVRC [6]~~[5]~~, EVRC-B [6] and VMR-WB [7]) produce packets in this format: 2 bytes for the encoded frame rate, followed by the encoded payload.

Table 6 Format of Encoded Speech File

Rate (2 octets)	Raw Speech Data (22 Octets)
4	Encoded Data
2	Encoded Data
…	…

~~**Table 6 Format of Encoded Speech File**~~

Output File:

Output File (to be used as input for Network Simulator): The format of the output file is described in Figure 8 with Time Stamp set to the creation time of the packet. In addition to the RTP packet, two additional header fields are required by the Output File, Packet size: size of the RTP packet (payload + header) and Time Stamp: time (in ms) at which the packet is created or received.

RTP packet size (4 bytes)	Time Stamp (in msec, 4 bytes)	RTP Header (12 bytes)	RTP Payload

Figure 8 Packet of Network Simulator input / output file

5.3.4.2 Network Simulator Input/Output for VoIP Testing

The Network Simulator takes the “delay profile” and input packet file as input, drops packets, changes time stamps, and produces an output packet file. In the case of drops, the output packet file does not contain dropped packets. For packets not dropped, the arrival time field will be modified based on the input delay profile. The output file is then fed into NCIM.

The input and output files of Network Simulator are as follows:

Input Files:

Delay Profile: As described in Section 4.2.3

Input File: The format of the input file is described in Figure 8 with Time Stamp set to the creation time of the packet.

Output File:

Output File (to be used as input for NCIM block): the format of the output file is described in Figure 8 with Time Stamp set to the arrival time of the packet. The Time Stamps in the output packets are updated based on input creation Time Stamps and delay profiles.

5.3.4.3 Network Client Interface Module (NCIM) for VoIP

NCIM simulates (i) the non-periodic arrival of RTP packets to a VoIP application and (ii) the periodic output of decoded speech from the VoIP client receiver (this is the Media Client Receiver for VoIP). To achieve these, NCIM passes the RTP packets which are extracted (without RTP packet size and Time Stamp) from Network Simulator Packets according to their arrival times to the VoIP client receiver and receives output PCM from the VoIP client receiver. To better understand the above proposed methodology, the interaction between the NCIM and the VoIP client receiver is illustrated in Figure 9.

The VoIP client receiver includes a speech decoder and potentially a jitter buffer management scheme (or schemes). The functionality of NCIM ensures that the VoIP client receiver receives packets in the manner defined in the delay profile.

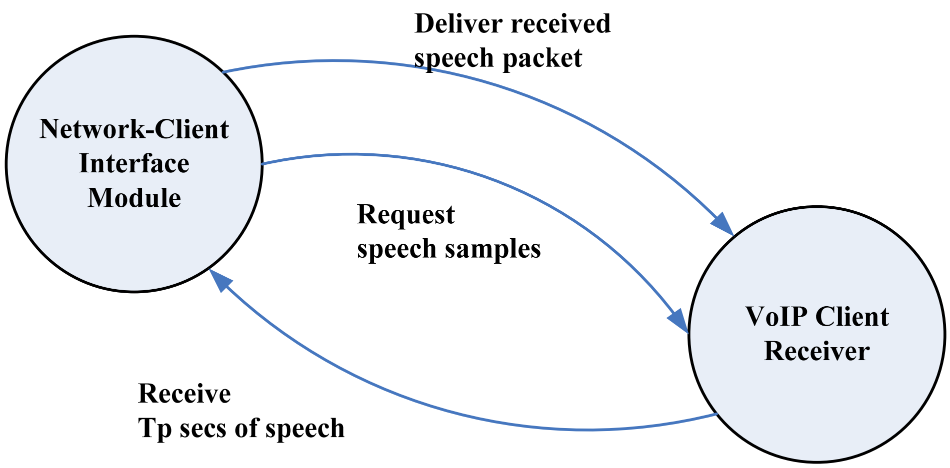

Figure 9 Testing Methodology for VoIP Codecs

NCIM Input/Output

The inputs and outputs of NCIM are described below:

Input:

NCIM takes as input the Network Simulator packet file outputted by Network Simulator with Time Stamp set to the arrival time of the packet. An example of the Input File is shown in Figure 10. In this figure, RTP packets are shown with two additional header fields, RTP packet size: size of the RTP packet (payload + header) and Arrival time: time (in ms) at which the packet is received.

PCM samples received from VoIP client receiver

RTP packet size 34	Arrival Time 120	RTP Header SN=0 TS=0	RTP Payload
RTP packet size 22	Arrival Time 150	RTP Header SN=1 TS=160	RTP Payload
RTP packet size 22	Arrival Time 175	RTP Header SN=2 TS=320	RTP Payload
RTP packet size 34	Arrival Time 185	RTP Header SN=3 TS=480	RTP Payload
RTP packet size 14	Arrival Time 200	RTP Header SN=4 TS=640	RTP Payload

Figure 10 Descriptive Example NCIM Input File

Output:

RTP packets: these are input to the VoIP client receiver

PCM Output for playout

NCIM Function and VoIP Client Receiver Interaction

In order to evaluate VoIP clients, NCIM uses the following methodology.

Note: In the pseudo-code below, NCIM uses the codec interface, the functionality of which is to be provided by the VoIP Client Receiver. This interface consists of the functions Codec.Init, Codec.InputPacket, Codec.OutputPCM and Codec.BufferFullness.

```
// Initialize the simulation loop

CurrentTime = 0
NextPacket = 0
NextPlayoutTime = Packets[0].ArrivalTime + InitialPlayoutDelay
SimulationFinished = FALSE

// Initialize the codec

Codec.Init()

// Process all received packets and output decoded speech

while (SimulationFinished = FALSE)
{
    // Receive packets
    while (Packets[NextPacket].ArrivalTime <= NextPlayoutTime)
    {
        // Feed packet to the codec
        Codec.inputPacket(Packets[NextPacket]);
        // Check if this was the last packet
        if (this was the last packet in the network trace)
```

```
            {
                SimulationFinished = TRUE
            }
            else
            {
                NextPacket = NextPacket + 1
            }
        }
        // Playout audio
        while ( (NextPlayoutTime < Packets[NextPacket].ArrivalTime)
                  or (SimulationFinished and Codec.BufferFullness>0))
        {
            // Request one block of audio from the codec
            //Audio: PCM structure and Codec.OutputPCM() stores PCM samples as file
            Codec.OutputPCM(Audio)
            // Estimate next playout time
            NextPlayoutTime = NextPlayoutTime + Audio.Duration
        }
    }
```

It should be noted that codec.inputPacket() may be called many times between two successive calls of codec.outputPCM(), and vice versa. Also, it should be noted that the Arrival Time field is an integer and the allowed time granularity is in multiples of 1 msec. The reason for this is that at every packet arrival time, the VoIP client receiver delivers PCM samples equivalent to the difference between the current packet's and the previous packet's arrival times. Due to this, this difference has to be a multiple of a PCM sample's equivalent time. In the case of 8kHz sampling, a PCM sample is equivalent to 1/8 msec. For other sampling rates, this may be different. Thus, for simplicity, it is assumed that the Arrival Time is an integer number of msec. This assumption simplifies the simulator for use with different codecs, sampling rates, etc., and does not adversely affect the accuracy of simulations.

5.3.4.4 Output of NCIM and VoIP Client Receiver

Output of VoIP Client:

An output file containing RTP Sequence Number, number of PCM samples and delay added by the VoIP client receiver for each played out packet. The delay added by the VoIP client receiver is the difference between the time the packet is played out and the time the packet was input to the VoIP client receiver. This output file will be produced by the VoIP client receiver and will have the format shown in Table 7 (in this format, the RTP Sequence Number will be set to 'E' for Erasure and 'S' for packets used by decoder to reconstruct background noise).

Raw PCM: the output PCM is passed from the VoIP client receiver to NCIM

Output of NCIM

Raw PCM file: NCIM generates a PCM output file at the end of the simulation

RTP packet: this is input to the VoIP client receiver

The reason for NCIM to output the PCM file rather than the VoIP client receiver doing it directly is so that VoIP client receivers are forced to produce PCM samples in simulated “real-time”, rather than only at the end of the simulation. If a VoIP client receiver is not asked to feed PCM samples to NCIM, then the VoIP client receiver can buffer up a large number of packets at the beginning and only then start producing PCM samples. This will allow the VoIP client to produce PCM files with high quality. Since it is not returning these PCM samples to NCIM at regular intervals, it can claim that the delays added by it are smaller than what the buffering up would allow. Returning PCM samples to NCIM, on the other hand, will mean that NCIM knows at every point what PCM samples were produced. This will enable NCIM to keep a check on the reported packet delays being correct. Therefore, this methodology enables fair evaluations of different jitter management schemes belonging to VoIP client receivers.

Table 7 Format of VoIP Client Output File

RTP Sequence Number	Number of PCM Samples	VoIP Client Receiver Delay (msec)
20	160	520
21	160	540
22	120	555
E	160	575
24	160	595

.

6 Quality Evaluation

In this section, the traditional metric (average PSNR) is addressed and then objective metrics that correlate better to human perception of video sequences in error-prone conditions are described. Note, the metrics presented here are non-exclusive and additional metrics can be used for evaluation.

6.1 Traditional Objective Measure

For a color video, the traditional method in video coding is to compute the average Peak Signal-to-Noise Ratio ($PSNR_{avg}$) of a luminance component as defined below.

$$dist_n = \Sigma_j \Sigma_k \ (x_{j,k}-y_{j,k})^2/N_{pel}$$

$$PSNR_{avg} = (\Sigma_n 10 \log_{10}(255 \text{ x } 255 / dist_n)) / N$$

where $x_{j,k}$ and $y_{j,k}$ are the <j,k> pixel values of the n^{th} original and reconstructed frames, respectively, N is the total number of frames, and N_{pel} is the total number of pixels for each frame. Some implementations use, Sum of Absolute Differences, SAD(a,b), instead due to complexity issues. $PSNR_{avg}$ has been used extensively in the field.

6.2 Objective Metrics For Error-prone Conditions

It is very well known that average *PSNR* does not correlate well with perceptual quality of reconstructed video sequences, particularly in an error prone environment. Three alternative objective measures are presented below.

6.2.1 Percentage Degraded Video Duration (pDVD)

The first proposed measure is based on the percentage of error-propagation intervals (corruption intervals) of the video sequence due to packet losses. First, average PSNR of each test video bitstream in an error-free environment is empirically measured. Next, packet errors are injected into the test video bitstreams using the network simulator and the error mask. First, define $t_i(1,x)$ as the starting point in time that the PSNR is dropped more than x dB compared to the PSNR at the same time when compared to the error-free case. Similarly, $t_i(2,x)$ is defined as an ending point in time when PSNR recovers to within x dB of the PSNR in an error-free condition. T_i is defined as the error-propagation interval, i.e., $T_i(x) = t_i(2,x) - t_i(1,x)$. If the ratio of the sum of all such error-propagation intervals to the entire sequence duration is calculated, a metric indicating the percentage of the clip that was degraded due to packet losses is provided. It is defined as,

$$\text{pDVD}(x) \quad = (\Sigma_n 1((PSNR_n^c - PSNR_n^e)>x)) / N$$

where, $PSNR_n^c$ and $PSNR_n^e$ are the PSNR of the n^{th} frame under error-free and error-prone conditions, respectively, $1(y)$ is 1 if y is true and 0 otherwise, x is a predefined threshold, and N is the number of frames in the video sequence. For example, as shown in Figure 11, Case B provides lower degraded video duration when compared to Case A, i.e., $pDVD_B < pDVD_A$. This metric is motivated from the fact that if the reconstructed clip is subject to longer corrupted durations, it will more severely affect the human-visual perception. Regardless of how good it was during un-corrupted duration, it may still be unsatisfactory.

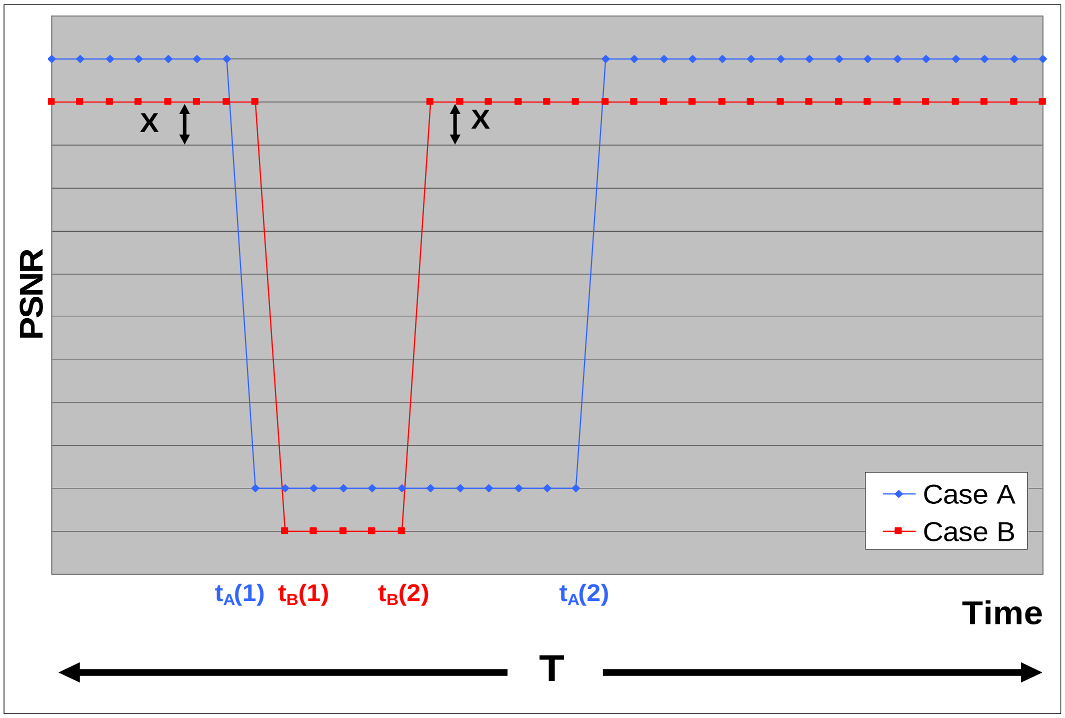

Figure 11 Example of PSNR Trace in Error-Prone Environment

6.2.2 Average PSNR in Clean and Error-Propagated Durations

Traditional objective measure computes the average PSNR from the entire duration. A different measure is to calculate two separate average PSNRs. First is the average PSNR during the total degraded video duration and the other is the average PSNR during the rest of the time as shown below.

$$PSNR_{avg}(Error) = (\Sigma_e\ 10 \log_{10}(255 \times 255 / dist_e)) / N_e$$

$$PSNR_{avg}(Clean) = (\Sigma_c\ 10 \log_{10}(255 \times 255 / dist_c)) / N_c$$

where e and c are defined as the frames corresponding to degraded video duration(s) and the error free video duration(s) as defined in 6.2.1. Furthermore, N_e and N_c are the number of frames corresponding to degraded video duration and error free video duration, respectively. For example, for Case A, e is from $t_A(1,x)$ to $t_A(2,x)$ while c is the duration “not” between $t_A(1,x)$ to $t_A(2,x)$. These objective metrics provide better insight on the received video quality by computing PSNR in both good and bad durations separately, rather than considering one global average PSNR.

6.2.3 Standard Deviation of PSNR

The final proposed metric is to compute the standard deviation of PSNR as shown below.

$$STD\_PSNR = sqrt\{(\Sigma_n (PSNR_n - PSNR_{avg})^2) / N\}$$

This measure gives another point of view to see how much PSNR varies during the entire sequence.

This page intentionally left blank.

7 Services

It is expected that relevant aspects of this document will be used during the development of multimedia service specifications and characterization documents. Shown in this section is an example of relevant parts of this document to be used for quality evaluation of cdma2000 multimedia services.

7.1 Packet Switched Video Telephony (PSVT)

Relevant parts of this document for PSVT simulation are Section 4, 5, 6.1, 6.2.1, 6.2.2, 7, and 8.

7.2 Multimedia Streaming Service (MSS)

Relevant parts of this document for MSS simulation are Section 4, 5, 6.1, 6.2.1, 6.2.2, 7, and 8.

7.3 Broadcast Multicast Service (BCMCS)

Relevant parts of this document for BCMCS simulation are Section 4, 5, 6.3, 7, and 8.

7.4 Voice over IP (VoIP)

Relevant parts of this document for VoIP simulation are Section 4.2, 5.2, 6.2.3, and 7.