VOICE OVER IP

Audio transferring throught the IP network

Measurement-Guide

25th Feb. 2009.

TECHNICAL UNIVERSITY OF BUDAPEST
Dept. of Telecommunications and Telematics

Content

1 Speech coding

Speech coding is the application of data compression of digital audio signals containing speech. Speech coding uses speech-specific parameter estimation using audio signal processing techniques to model the speech signal, combined with generic data compression algorithms to represent the resulting modeled parameters in a compact bitstream.

The two most important applications of speech coding are mobile telephony and Voice over IP.

The techniques used in speech coding are similar to that in audio data compression and audio coding where knowledge in psychoacustics is used to transmit only data that is relevant to the human auditory system. For example, in narrowband speech coding, only information in the frequency band 400 Hz to 3500 Hz is transmitted but the reconstructed signal is still adequate for intelligibility.

Speech coding differs from other forms of audio coding in that speech is a much simpler signal than most other audio signals, and that there is a lot more statistical information available about the properties of speech. As a result, some auditory information which is relevant in audio coding can be unnecessary in the speech coding context. In speech coding, the most important criterion is preservation of intelligibility and "pleasantness" of speech, with a constrained amount of transmitted data.

It should be emphasised that the intelligibility of speech includes, besides the actual literal content, also speaker identity, emotions, intonation, timbre etc. that are all important for perfect intelligibility. The more abstract concept of pleasantness of degraded speech is a different property than intelligibility, since it is possible that degraded speech is completely intelligible, but subjectively annoying to the listener.

In addition, most speech applications require low coding delay, as long coding delays interfere with speech interaction.

- Speech coding features

1. **Delay or latency:** VoIP delay or latency is characterized as the amount of time it takes for speech to exit the speaker and reach the listener. The ITU-T recommendation specifies that for good voice quality, no more than 150 ms of one-way, end-to-end delay should occur. In an unmanaged, congested network, queing delay can add up to two seconds of delay. This lenghty period of delay is unacceptable in almost any voice network. Queuing delay is only one component of end-to-end delay. A packet-based networks generate various delay for various reasons.The main parts of the total delay are following: propagation delay, handling delay, queuing delay and jitter.

2. **Bit rate and required bandwidth:** The bit rates of the coders defined by the ITU range from the low 2.4 kbit/s coders used in secure telephony to 64 kbit/s wideband

coders, such as the G.722 or the G.711 pulse-code modulated (PCM) coder. The rate of the coder determines the required channel bandwidth. In cellular telephony, for instance, preserving bandwidth is crucial. As such, variable bit rate coders, such the enhanced variable rate coder (EVRC) used in 2G CDMA systems were designed to drop the coding rate during speech inactivity.

3. **Complexity:** Speech coding algorithms are in general computation intensive. As a result, they are typically implemented on programmable digital signal processors (DSPs) that are optimized for signal processing operations, such as convolutions, Fast Fourier transforms (FFTs), and digital filtering. PC-based processors have in recent years evolved to provide enough processing power to make them appropriate candidates to run complex operations such as speech coding. As the VLSI technology enables more MIPS per silicon area, at a decreasing cost, the complexity aspect is less crucial than it used to be. However, it is always desirable to pack as much functionality in a processor, and have efficient algorithms that do not use up a large percentage of the available processing power.

The commonly used coders such as G.723, G.729, and G.728 were developed with specific requirements and priorities in mind; as such, they provide different levels of compromises along these four dimensions. (Table 1).

Table 1: Summary of Attributes for 3 Commonly Used VoIP Coders

Attribute	G.723.1	G.729	G.729a
Bit rate	6.4kbit/s 5.33kbit/s	8 kbit/s	8 kbit/s
Frame size	30 ms	10 ms	10 ms
Look ahead	7.5 ms	5 ms	5 ms
Total delay	67.5 ms	25 ms	25 ms
Complexity RAM	16 MIPS 2.2 kwords	20 MIPS 3 kwords	10 MIPS 2 kwords

4. **Quality:** To ensure high speech quality in an IP network, designers must realize that many of the challenges lie in the inherent nature of the network. There are really five big issues that designers will encounter in the network: packet loss/bit-error rates, delay, jitter, echo background noise, and tandeming effects.

Commonly used is the Mean Opinion Score (MOS) value based on the ITU-T P.800 standard. MOS gives a numerical indication of the perceived quality of the media received after being transmitted and eventually compressed using codecs.

MOS is expressed in one number, from 1 to 5, 1 being the worst and 5 the best. MOS is quite subjective, as it is based figures that result from what is perceived by people during tests. However, there are software applications that measure MOS on networks.

The Mean Opinion Score Values: Taken in whole numbers, the numbers are quite easy to grade: 5 – Excellent, 4 – Good, 3 – Fair, 2 – Poor, 1 – Bad.
5 - Perfect. Like face-to-face conversation or radio reception.

4 - Fair. Imperfections can be perceived, but sound still clear. This is (supposedly) the range for cell phones.
3 - Annoying.
2 - Very annoying. Nearly impossible to communicate.
1 - Impossible to communicate

PSQM (***Perceptual Speech Quality Measure***) is a computational and modeling algorithm defined in ITU Recommendation ITU-T P.861 that objectively evaluates and quantifies voice quality of voice-band (300 - 3400 Hz) speech codecs. It may be used to rank the performance of these speech codecs with differing speech input levels, talkers, bit rates and transcodings. The ITU-T has Withdrawn P.861 and replaced it with P.862 (PESQ) which contains an improved speech assessment algorithm.

Using the PSQM standard allows automated, simulation-based test methodologies to objectively rate both speech clarity and transmitted voice quality.

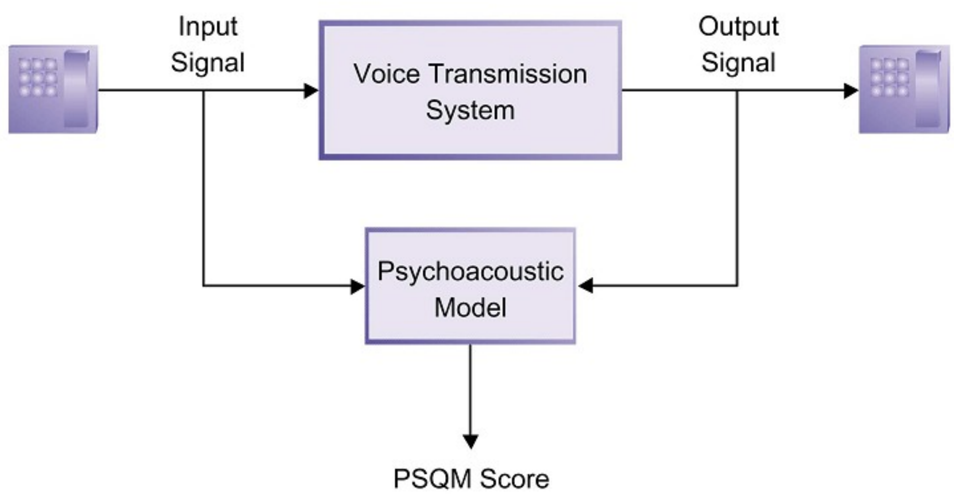


Figure 1 PSQM system

PSQM as originally conceived was not developed to account for network QoS perturbations common in VoIP applications, items such as packet loss, delay variance (jitter) or non-sequential packets. These conditions usually give inappropriate results under heavy network load simulations, failing to account for a very real perceived loss of voice quality. Attempts to duplicate network fault conditions by introducing significant packet loss result in PSQM values that correspond to falsely inflated MOS values.

In order to overcome this limitation, PSQM+ was developed by modifying the original algorithm. PSQM+ generates results that seem to more accurately reflect the adverse performance of speech codecs under realistic network load conditions.

2 Standard codecs

- G.711 (PCM)

G.711 is an ITU-T standard for audio companding. It is primarily used in telephony. The standard was released for usage in 1972. G.711 represents logarithmic Pulse-Code Modulation (PCM) samples for signals of voice frequencies, sampled at the rate of 8000 samples/second.

There are two main compression algorithms defined in the standard, the u-law algorithm(used in North America & Japan) and A-law algorithm (used in Europe and the rest of the world). Both are logarithmic, but A-law was specifically designed to be simpler for a computer to process. The standard also defines a sequence of repeating code values which defines the power level of 0 dB.

The μ-law and A-law algorithms encode 14-bit and 13-bit signed linear PCM samples (respectively) to logarithmic 8-bit samples. Thus, the G.711 encoder will create a 64 kbit/s bitstream for a signal sampled at 8 kHz.

G.711, also known as Pulse Code Modulation (PCM), is a very commonly used waveform codec. G.711 uses a sampling rate of 8,000 samples per second, with the tolerance on that rate 50 parts per million (ppm). Non-uniform quantization with 8 bits is used to represent each sample, resulting in a 64 kbit/s bit rate. There are two slightly different versions; μ-law, which is used primarily in North America, and A-law, which is in use in most other countries outside North America. G.711 μ-law tends to give more resolution to higher range signals while G.711 A-law provides more quantization levels at lower signal levels. When using μ-law G.711 in networks where suppression of the all 0 character signal is required, the character signal corresponding to negative input values between decision values numbers 127 and 128 should be 00000010 and the value at the decoder output is -7519. The corresponding decoder output value number is 125.

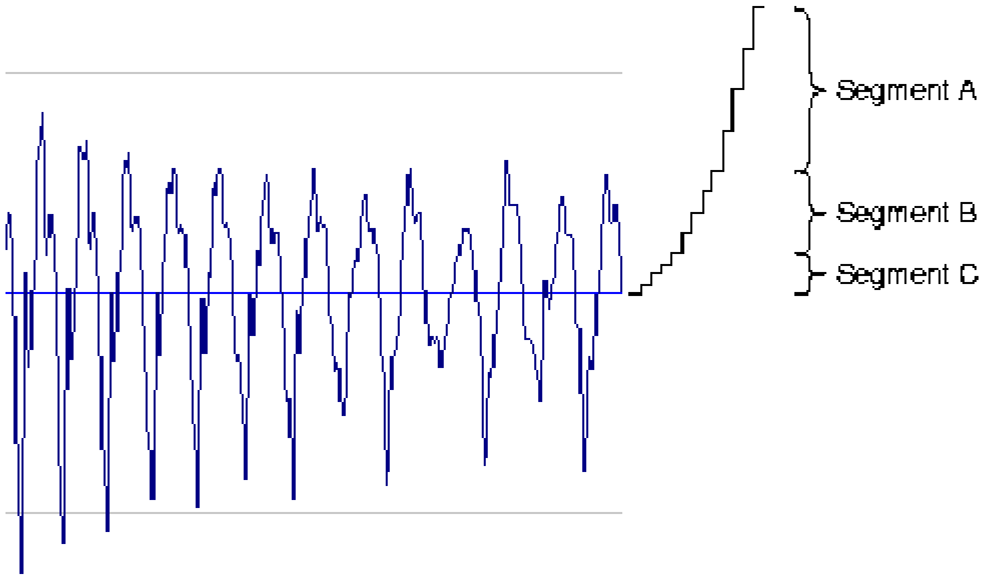


Figure 2 The u-law coding

We can express the formula for the u-law coding by the next way. Commonly 12..16 bites long samples are coding, where the maximum value of the sample is: mp=2048 or 32768.

The u-LAW (mu-LAW):

$$y = \frac{\operatorname{sgn}(m)}{\ln(1+\mu)} \cdot \ln\left(1+\mu\cdot\left|\frac{m}{mp}\right|\right), \quad \left|\frac{m}{mp}\right| \leq 1$$

m is the current sample value, u =100 or 255.

Another form of the matemathic formula is:

$$y = \operatorname{sgn}(x)\cdot\frac{\ln(1+255\cdot|x|}{\ln(1+255)}$$

where

x = normalized input (value between -1 and 1)

255 = compression parameter

sgn(x) = sign (+/-) before x

The result conversion is on the next figure.

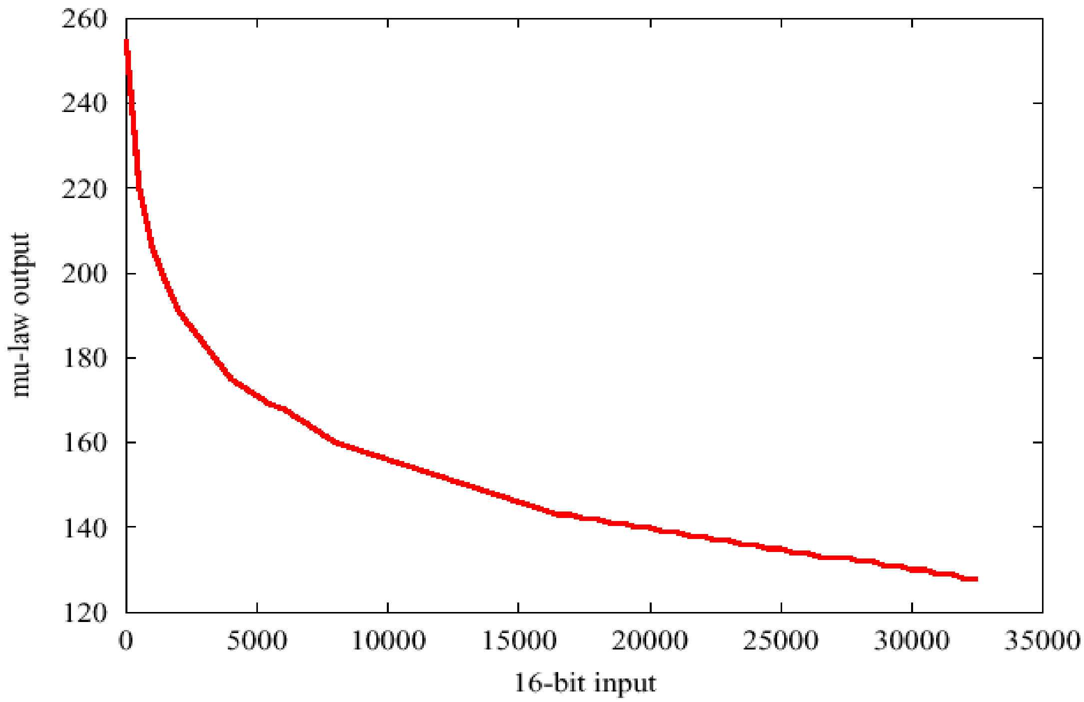


Figure 3 The u-law coding characteristic

The a-law definition is:

$$y = \begin{cases} \dfrac{A}{1+\ln(A)}\left(\dfrac{m}{mp}\right), & if \quad \left(\dfrac{m}{mp}\right) \le \dfrac{1}{A} \\ \dfrac{\operatorname{sgn}(m)}{1+\ln(A)} \cdot \left(1+\ln(A)\left(\dfrac{m}{mp}\right)\right), & if \quad 1 \le m \le 1 \end{cases}$$

Where A=87.6, mp – max. value of the sample, m = current sample value.

Features:

Bit rates [kbit/s]:	64
Frame-delay [ms]:	0.125
Frame-length [byte]:	1
Lookahead [ms]:	-
MOS values:	4.3-4.7
MIPS:	0.52

Advantages:

- simple, little komplexity
- small delay
- Good speech-quality

Disadvantages:

- Big bandwidth

- G.721 (ADPCM)

G.726 is an ITU-T Adaptive Differential Pulse Code Modulation (ADPCM) speech codec standard covering the transmission of voice at rates of 16, 24, 32, and 40 kbit/s. It was introduced to supersede both G.721, which covered ADPCM at 32 kbit/s, and G.723, which described ADPCM for 24 and 40 kbit/s. G.726 also introduced a new 16 kbit/s rate. The four bit rates associated with G.726 are often referred to by the bit size of a sample, which are 2-bits, 3-bits, 4-bits, and 5-bits respectively.

The most commonly used mode is 32 kbit/s, since this is half the rate of G.711, thus increasing the usable network capacity by 100%. It is primarily used on international trunks in the phone network. It also is the standard codec used in DECT wireless phone systems.

The simplified diagram of the ADPCM coder/decoder is on the next figure:

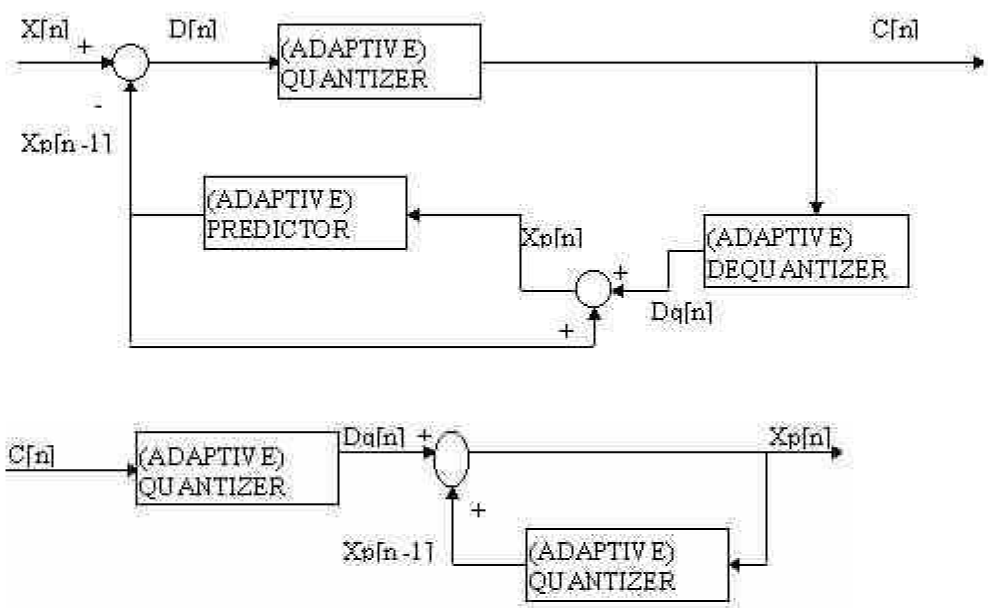


Figure 4 ADPCM coder/decoder

Short for ***A****daptive* ***D****ifferential* ***P****ulse* ***C****ode* ***M****odulation,* a form of *pulse code modulation (PCM)* that produces a digital signal with a lower bit rate than standard PCM. ADPCM produces a lower bit rate by recording only the difference between samples and adjusting the coding scale dynamically to accommodate large and small differences. Some applications use ADPCM to digitize a voice signal so voice and data can be transmitted simultaneously over a digital facility normally used only for one or the other.

Features:

Bit rates [kbit/s]: 32

Frame-delay [ms]:	0.125
Frame-length[byte]:	0.5
Lookahead [ms]:	-
MOS value:	4.0-4.2
MIPS:	7.2

Advantages:

- simple, little komplexity
- small delay
- Good speech-quality

Disadvantages:

- Relatively big bandwidth
- Fix coding rate

- G.722 (SB-ADPCM)

G.722 (ITU-T, AD-PCM: Sub-Band Adaptive Differential Pulse Code Modulation) is a midle quality speech-coding, a variant of the G.721 ADPCM coding. The coding system uses sub-band adaptive differential pulse code modulation (SB-ADPCM) within a bit rate of 64 kbps. The system is referred as 64 kbps (7 kHz) audio coding.
In the SB-ADPCM technique used, the frequency band is split into two sub-bands (higher and lower) and the signals in each sub-band are encoded using ADPCM. The system has three basic modes of operation corresponding to the bit rates used for 7 kHz audio coding: 64, 56 and 48 kbps. The latter two modes allow an auxiliary data channel of 8 and 16 kbps respectively to be provided within the 64 kbps by making use of bits from the lower sub-band.

Features:

Bit rates [kbit/s]:	48	56	64
Frame delay [ms]:	0.125	0.125	0.125
Frame length[byte]:	0.750	0.850	1
Lookahead [ms]:	1.5		
MOS value:	3.1-4.5		
MIPS:	10-13.9		

Advantages:

- simple, little komplexity
- small delay
- Good speech-quality
- Various coding rates

Disadvantages:

- Relatively big bandwidth

- G.722.2 (AMR-WB)

ITU-T G.722.2 is an international standard for wide-band audio compression. The G. 722.2 standard uses an Adaptive Multi-Rate Wide-Band (AMR-WB) codec. GAO's G. 722.2 multi-rate codec operates between 6.6 kbit/s to 23.85 kbit/s. By utilizing a wide-band of 50Hz to 7kHz, GAO's G.722.2 algorithm provides higher-than-toll quality speech. The algorithm utilizes ACELP technology. Both the IP network and the 3G network use the same codec which provides seamless interoperability.

Features:

Bit rates[kbit/s]:	23.85-6.6
Frame delay [ms]:	20
Frame length[byte]:	
Lookahead [ms]:	5
MOS value:	>4.0
MIPS:	40-45

Advantages:

- small delay
- very good speech-quality
- Various coding rates

Disadvantages:

- big bandwidth for the good quality speech
- middle komplexity

- G.723.1

G.723.1 specifies a coded representation that can be used for compressing the speech or other audio signal component of multimedia services at a very low bit rate. In the design of this coder, the principal application considered was very low bit rate visual telephony as part of the overall H.324 family of standards. G.723.1 has two bit rates associated with it, 5.3 kbit/s and 6.3 kbit/s. The higher bit rate has better voice quality. The lower bit rate still gives good quality and provides system designers with additional flexibility. Both rates are a mandatory part of the encoder and decoder, it is possible to switch between the two rates at any 30 ms frame boundary. An option for variable rate operation using discontinuous transmission and noise fill during non-speech intervals is also available.

Features:

Bit rates [kbit/s]:	6.4	5.3
Frame-delay [ms]:	30	30
Frame-length [byte]:	24	20
Lookahead [ms]:	7.5	
MOS value:	3.8-4	3.5-3.7

MIPS:	16.9	16.5

Advantages:

- small bandwidth
- two types of coding rates

Disadvantages:

- Relatively low speech-quality
- Relatively high complexity
- Big coding-time
- Need a license

- G.726 (ADPCM)

G.726 is an ITU-T Adaptive Differential Pulse Code Modulation (ADPCM) speech codec standard covering the transmission of voice at rates of 16, 24, 32, and 40 kbit/s. It was introduced to supersede both G.721, which covered ADPCM at 32 kbit/s, and G.723, which described ADPCM for 24 and 40 kbit/s. G.726 also introduced a new 16 kbit/s rate. The four bit rates associated with G.726 are often referred to by the bit size of a sample, which are 2-bits, 3-bits, 4-bits, and 5-bits respectively.

The most commonly used mode is 32 kbit/s, since this is half the rate of G.711, thus increasing the usable network capacity by 100%. It is primarily used on international trunks in the phone network. It also is the standard codec used in DECT wireless phone systems.

Features:

Bit rates [kbit/s]:	16	24	32	40
Frame-delay [ms]:	0.125	0.125	0.125	0.125
Frame-legth [byte]:	0.250	0.375	0.5	0.625
Lookahead [ms]:	-			
MOS value:	?	3.7	3.9	4.2
MIPS:	7.2-12			

Advantages:

- simple, little komplexity
- small delay
- Good speech-quality
- Various coding rates

Disadvantages:

- Relatively big bandwidth
- The speech-quality is worse on the low bandwidth

- G.728 (LD-CELP)

G.728 is a ITU-T standard for speech coding operating at 16 kbit/s. Technology used is LD-CELP, low-delay code excited linear prediction. Delay of the codec is only 5 samples (0.625 ms). The linear prediction is calculated backwards with a 50th order LPC filter. The excitation is generated with gain scaled VQ. The standard was finished in 1992 in the form of algorithm exact floating point code. In 1994 a bit exact fixed point codec was released. G.728 passes low bit rate modem signals up to 2400 bit/s. Also network signaling goes through. The complexity of the codec is 30 MIPS. 2 kilobytes of RAM is needed for codebooks.

Features:

Bit rates [kbit/s]:	12.8	16
Frame-delay [ms]:	0.625	0.625
Frame-length [byte]:	1	1.250
Lookahead [ms]:	-	
MOS value:	3.7	4.3
MIPS:	16-35	

Advantages:
- small delay
- Good speech-quality

Disadvantages:
- Relatively big bandwidth (16kbp/s)
- Fix coding rate
- Noice sensitive

- G.729 (CS-ACELP)

The G.729 speech coder is an 8 kbps Conjugate-Structure Algebraic-Code-Excited Linear Prediction (CS-ACELP) speech compression algorithm approved by ITU-T. G.729 offers high quality, robust speech performance at the price of complexity. It requires 10 ms input frames and generates frames of 80 bits in length. With the G.729 coder processing signals in 10 ms frames and a 5 ms look-ahead, the total algorithmic delay is 15 ms.

Features:

Bit rate [kbit/s]:	8
Frame-delay [ms]:	10
Frame-length [byte]:	10
Lookahead [ms]:	5
MOS value:	3.9-4.2
MIPS:	10-11

Advantages:
- Acceptable komplexity
- Good speech-quality
- Relatively small delay
- good compression-rate (8kbit/s)

Disadvantages:
- fix coding-rate
- need license

- GSM 06.10 (GSM)

GSM 06.10 FR Vocoder defines a reference configuration for the speech transmission chain of the digital cellular telecommunications system. The speech encoder takes its input as a 13 bit uniform PCM signal either from the audio part of the mobile station or on the network side, from the PSTN via an 8 bit/A-law to 13 bit uniform PCM conversion. The encoded speech at the output of the speech encoder is delivered to a channel encoder unit which is specified in GSM 05.03. In the receive direction, the inverse operations take place.

GSM 06.10 describes the detailed mapping between input blocks of 160 speech samples in 13 bit uniform PCM form to encoded blocks of 260 bits and from encoded blocks of 260 bits to output blocks of 160 reconstructed speech samples. The sampling rate is 8000 sample/s leading to an average bit rate for the encoded bit stream of 13 kbit/s. The coding scheme is the so-called Regular Pulse Excitation - Long Term prediction - Linear Predictive Coder, here-after referred to as RPE-LTP.

The GSM coder output is a good speech-quality, but the G.728 coder (CELP) need higher bandwidth. The GSM's complexity is low.

Features:

Bit rates [kbit/s]:	13	7
Frame-delay [ms]:	20	20
Frame-length [byte]:	10	10
Lookahead [ms]:	-	
MOS value:	3.5-3.7	
MIPS:	4.5	

Advantages:
- Simple, relatively small complexity
- small bandwidth
- small delay
- open source

Disadvantages:

- the bandwidth/speech-quality rate is not so good

- AMR-NB

Adaptive Multi-Rate (AMR) is a patented audio data compression scheme optimized for speech coding. AMR was adopted as the standard speech codec by 3GPP in October 1998 and is now widely used in GSM and UMTS. It uses link adaptation to select from one of eight different bit rates based on link conditions.

AMR is also a file format for storing spoken audio using the AMR codec. Many modern mobile telephone handsets will allow you to store short recordings in the AMR format, both Open Source (see the external links) and commercial programs exist to convert between this and other formats such as MP3, although it should be remembered that AMR is a speech format and is unlikely to give ideal results for other audio.

There are a total of 14 modes of the AMR codec, 8 are available in a full rate channel (FR) and 6 on a half rate channel (HR).

Mode	**Bitrate (kbit/s)**	**Channel**	**Compatible with**
AMR_12.20	12.20	FR	ETSI GSM enhanced full rate
AMR_10.20	10.20	FR	
AMR_7.95	7.95	FR/HR	
AMR_7.40	7.40	FR/HR	TIA/EIA IS-641 TDMA enhanced full rate
AMR_6.70	6.70	FR/HR	ARIB 6.7 kbit/s enhanced full rate
AMR_5.90	5.90	FR/HR	
AMR_5.15	5.15	FR/HR	
AMR_4.75	4.75	FR/HR	
AMR_SID	1.80	FR/HR	

Features:

Bit rate [kbit/s]:	4.75-12.2
Frame-delay [ms]:	20
Frame-length [byte]:	10
Lookahead [ms]:	5
MOS value:	3.5-4.01
MIPS:	15-25

Advantages:

- Simple, relatively small complexity
- Small bandwidth
- Small delay
- Good speech-quality
- Various coding rate

Disadvantages:

- only few implemantations exist
- doesn’t exist open source

- Summary table

Codec	Bandwidth	Frame-delay	Frame-length	Lookahead	MOS value	MIPS
G.711	64.00	0.125	1.000	0.00	4.3 - 4.7	0.52
G.711	56.00	0.125	0.875	0.00	4.1*	0.52
G.711	48	0.125	0.750	0.00	3.7-3.9*	0.52
G.721	32	0.125	0.500	0.00	4.0 - 4.2	7.2
G.722	48.00	0.125	0.750	1.50	3.16 - 4.5	10 - 13.9
G.722	56.00	0.125	0.875	1.50	4.5	10 - 13.9
G.722	64.00	0.125	1.000	1.50	3.75 - 4.5	10 - 13.9
G.722.1	16.00	20.000	40.000	20	3.5-3.8	7.2 - 12
G.722.1	24.00	20.000	60.000	20.00	4.1	7.2 – 12
G.722.1	32.00	20.000	80.000	20.00	4.0	7.2 – 12
G.723.1	5.30	30.000	20.000	7.50	3.5 - 3.7	16.5
G.723.1	6.40	30.000	24.000	7.50	3.8 - 4.0	16.9
G.726	16.00	0.125	0.250	0.00	3.5-3.8	7.2 - 12
G.726	24.00	0.125	0.375	0.00	3.7	7.2 - 12
G.726	32.00	0.125	0.500	0.00	3.9 - 4.2	7.2 - 12
G.726	40.00	0.125	0.625	0.00	3.9	7.2 - 12
G.727	16.00	0.125	0.250	0.00	3.5-3.8*	9.9
G.727	24.00	0.125	0.375	0.00	3.7	9.9
G.727	32.00	0.125	0.500	0.00	3.9 - 4.2	9.9
G.727	40.00	0.125	0.625	0.00	3.9	9.9
G.728	12.80	0.625	1.000	0.00	3.5-4.1	16
G.728	16.00	0.625	1.250	0.00	3.7 - 4.3*	16 - 35
G.729	8.00	10.000	10.000	5.00	3.9 - 4.2	20 - 25
G.729a	8.00	10.000	10.000	5.00	3.7 - 4.2	10 - 11.4
G.729b	8.00	10.000	10.000	5.00	3.9 - 4.2	G.729 + .7
G.729ab	8.00	10.000	10.000	5.00	3.7 - 4.2	G.729a + .8
G.729d	6.40	10.000	8.000	5.00	3.5 – 3.9	< G.729
G.729e	8.00	10.000	10.000	5.00	4.0 - 4.2	30
G.729e	11.80	10.000	14.750	5.00	4.0 - 4.2	30
GSM	13.00	20.000	10.00	0	3.7	4.5
GSM-HR	7.00	20.000	10.00	0	3.5	4.5
AMR-NB	4.75	20.000	N.D	5.00	3.75	15
AMR-NB	12.2	20.000	N.D	5.00	4.01	25

N.D – No Data. * - estimated value

3 Az RTP protocol

The Real-time Transport Protocol (or RTP) defines a standardized packet format for delivering audio and video over the Internet.
RTP was originally designed as a multicast protocol, but has since been applied in many unicast applications. For host-to-host transport, RTP and RTCP use the User Datagram Protocol (UDP) predominantly, although other Transport Layer protocols, in particular Datagram Congestion Control Protocol (DCCP) and Stream Control Transmission Protocol (SCTP) may be used because of their congestion control mechanisms. Transmission Control Protocol (TCP) has been documented for RTP usage, but is rarely deployed in such applications. Applications using RTP are less sensitive to packet loss, but typically very sensitive to delays resulting from network latency, so UDP is a better choice than TCP for such applications.
The RTP is a protocol for the real-time traffic service from peer to peer. Real-time traffic services are the speech and the video.

The services provided by RTP include:

- Payload-type identification - Indication of what kind of content is being carried
- Sequence numbering - PDU sequence number
- Time stamping - allow synchronization and jitter calculations
- Delivery monitoring

The protocols themselves do not provide mechanisms to ensure timely delivery. They also do not give any Quality of Service (QoS) guarantees. These things have to be provided by some other mechanism.

Also, out of order delivery is still possible, and flow and congestion control are not supported directly. However, the protocols do deliver the necessary data to the application to make sure it can put the received packets in the correct order. Also, RTCP provides information about reception quality which the application can use to make local adjustments. For example if a congestion is forming, the application could decide to lower the data rate.

The RTP packet header structure is fix:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Ver|P|X|  CC   |M|     PT      |     Sequence Number           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Timestamp                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           szinkronizációs forrás (SSRC)azonosító              |
+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
|            közreműködő forrás (CSRC) azonosító                |
|                             ....                              |
```

The RTP header has a minimum size of 12 octets.

Ver.
: (2 bits) Indicates the version of the protocol. Current version is 2.

P (Padding)
: (1 bit) Used to indicate if there are extra padding bytes at the end of the RTP packet.

X (Extension)
: (1 bit) Indicates presence of an *Extension header* between standard header and payload data.

CC (CSRC Count)
: (4 bits) Contains the number of CSRC identifiers that follow the fixed header.

M (Marker)
: (1 bit) Used at the application level and is defined by a profile. If it is set, it means that the current data has some special relevance for the application.

PT (Payload Type)
: (7 bits) Indicates the format of the payload and determines its interpretation by the application.

Sequence Number
: (16 bits) The sequence number increments by one for each RTP data packet sent, and may be used by the receiver to detect packet loss and to restore packet sequence.

Timestamp
: (32 bits) The timestamp reflects the sampling instant of the first data in the RTP data packet. The sampling instant MUST be derived from a clock that increments monotonically and linearly in time to allow synchronization and jitter calculations. The resolution of the clock MUST be sufficient for the desired synchronization accuracy and for measuring packet arrival jitter (one tick per video frame is typically not sufficient). The clock frequency is dependent on the format of data carried as payload and is specified statically in the profile or payload format specification that defines the format, or MAY be specified dynamically for payload formats defined through non-RTP means. If RTP packets are generated periodically, the nominal sampling instant as determined from the sampling clock is to be used, not a reading of the system clock.By default the value is 0 which means it is infinite

SSRC

(32 bits) Synchronization source identifier uniquely identifies the source of a stream.

CSRC

Contributing source IDs enumerate contributing sources to a stream which has been generated from multiple sources.

Extension header

The first 32-bit word contains a profile specific identifier (16 bits) and a length specifier (16 bits) that indicates the length of the extension (EHL=extension header length) in 32-bit units, excluding the 32 bits of the extension header.

The RTP can do the following functions by these header fields:

- Time recovering (időbélyeg mező)
- Source identification (SSRC)
- Payload (content) identification (PT)
- Sequence-numbering
- Packet-lost recognizing
- Next functions are out of the RTP scope:
- QoS (quality of service) , resource management
- Packet-lost recovering

RTP is usually transmitted in packet sizes of multiples of 10 ms duration, but packets can be as small as 5 ms with some 16 kHz voice encoders. This results in a relatively large ratio of the size of the sum of RTP, UDP and IP headers and the size of the actual media data carried in a single packet. Efforts exist in Compressed RTP (CRTP) which reduces the size of the IP, UDP and RTP headers. It is primarily used for reliable and fast point-to-point links, but it can be problematic in other applications. Therefore, Enhanced CRTP (ECRTP) was defined.

Especially in VoIP over wireless applications, headers are significantly larger than the payload.

RTP allows only one type of message, one that carries data from the source to the destination. In many cases, there is a need for other messages in a session. These messages control the flow and quality of data and allow the recipient to send feedback to the source or sources. Real-time transport control protocol(RTCP) is a protocol designed for this purpose.

RTCP has five types of messages: sender report, receiver report, source description message, bye message, application-specific message.

Sender Report

The sender report is sent periodically by the active senders in a conference to report transmission and reception statistics for all RTP packets sent during the interval. The sender report includes an absolute timestamp, which is the number of seconds elapsed since midnight on January 1, 1900. The absolute timestamp allows the receiver to sychronize different RTP messages. It is particularly important when both audio and video are transmitted (audio and video transmissions use separate relative timestamps).

Receiver Report

The receiver report is for passive participants, those that do not send RTP packets. The report informs the sender and other receivers about the quality of service.

Source Description Message

The source periodically sends a source description message to give additional information about itself. This information can be the name, e-mail address, telephone number, and address of the owner or controller of the source.

Bye Message

A source sends a bye message to shut down a stream. It allows the source to announce that it is leaving the conference. Although other sources can detect the absence of a source, this message is a direct announcement. It is also very useful to a mixer.

Application-Specific Message

The application-specific message is a packet for an application that wants to use new applicatons (not defined in the standard). It allows the definition of a new message type.

Extra headers by the packet:

- RTP: 12 byte
- UDP: 8 byte
- IP: 20 byte
- Summary 40 byte.

The actual bandwidth dependences on the codec type and the sample duration by the packet. The bandwith strongly dependences on this sample duration by the packet.
Shorter duration -> bigger bandwidth is needed, longer duration -> bigger delay.
The sample duration value typically = 20 msec.

4 SIP signalling protocol

The Session Initiation Protocol (SIP) is a signalling protocol, widely used for setting up and tearing down multimedia communication sessions such as voice and video calls over the Internet. Other feasible application examples include video conferencing, streaming multimedia distribution, instant messaging, presence information and online games.

The protocol can be used for creating, modifying and terminating two-party (unicast) or multiparty (multicast) sessions consisting of one or several media streams. The modification can involve changing addresses or ports, inviting more participants, adding or deleting media streams, etc.

SIP was originally designed by Henning Schulzrinne and Mark Handley starting in 1996. The latest version of the specification is RFC 3261 from the IETF SIP Working Group. In November 2000, SIP was accepted as a 3GPP signaling protocol and permanent element of the IMS architecture for IP-based streaming multimedia services in cellular systems.

The SIP protocol is situated at the session layer in the OSI model, and at the application layer in the TCP/IP model. SIP is designed to be independent of the underlying transport layer; it can run on TCP, UDP, or SCTP. SIP has the following characteristics:

- Transport-independent, because SIP can be used with UDP, TCP, SCTP, etc.

- Text-based – *HTTP feature*, allowing for humans to read and analyze SIP messages.

SIP clients typically use TCP or UDP (typically on port 5060 and/or 5061) to connect to SIP servers and other SIP endpoints. SIP is primarily used in setting up and tearing down voice or video calls. However, it can be used in any application where session initiation is a requirement. These include Event Subscription and Notification, Terminal mobility and so on. There are a large number of SIP-related RFCs that define behavior for such applications. All voice/video communications are done over separate session protocols, typically RTP.

A motivating goal for SIP was to provide a signaling and call setup protocol for IP-based communications that can support a superset of the call processing functions and features present in the public switched telephone network (PSTN). SIP by itself does not define these features; rather, its focus is call-setup and signaling. However, it was designed to enable the construction of functionalities of network elements designated Proxy Servers and User Agents. These are features that permit familiar telephone-like operations: dialing a number, causing a phone to ring, hearing ringback tones or a busy signal. Implementation and terminology are different in the SIP world but to the end-user, the behavior is similar.

SIP-enabled telephony networks can also implement many of the more advanced call processing features present in Signaling System 7 (SS7), though the two protocols themselves are very different. SS7 is a centralized protocol, characterized by a complex central network architecture and dumb endpoints (traditional telephone handsets). SIP is a peer-to-peer protocol, thus it requires only a simple (and thus scalable) core network with intelligence distributed to the network edge, embedded in endpoints (terminating devices built in either hardware or software). SIP features are implemented in the communicating endpoints (i.e. at the edge of the network) contrary to traditional SS7 features, which are implemented in the network.

Although several other VoIP signaling protocols exist, SIP is distinguished by its proponents for having roots in the IP community rather than the telecom industry. SIP has been standardized and governed primarily by the IETF while the H.323 VoIP protocol has been traditionally more associated with the ITU. However, the two organizations have endorsed both protocols in some fashion.

SIP works in concert with several other protocols and is only involved in the signaling portion of a communication session. SIP is a carrier for the Session Description Protocol (SDP), which describes the media content of the session, e.g. what IP ports to use, the codec being used etc. In typical use, SIP "sessions" are simply packet streams of the Real-time Transport Protocol (RTP). RTP is the carrier for the actual voice or video content itself.

The first proposed standard version (SIP 2.0) was defined in RFC 2543. The protocol was further clarified in RFC 3261, although many implementations are still using interim draft versions. Note that the version number remains 2.0.

SIP is similar to HTTP and shares some of its design principles: It is human readable and request-response structured. SIP proponents also claim it to be simpler than H.323. However, some would counter that while SIP originally had a goal of simplicity, in its current state it has become as complex as H.323. Others would argue that SIP is a stateless protocol, hence making it possible to easily implement failover and other features that are difficult in stateful protocols like H.323. SIP and H.323 are not limited to voice communication and can mediate any kind of communication session e.g. voice, video and yet to be defined future formats

- ***SIP network elements***

SIP User Agents (UAs) are the end-user devices, used to create and manage a SIP session. A SIP UA has two main components, the User Agent Client (UAC) sends messages and answers with SIP responses, the User Agent Server (UAS) responds to SIP requests sent by the peer. SIP UAs may work in point to point mode. Typical implementations of a UA are SIP softphones, SIP hardphones and SIP-enabled ATAs.

In SIP, as in HTTP, the User Agent may identify itself using a string 'User-Agent', containing a text description of the software/hardware/product involved. In the case of SIP, this User-Agent field is sent along on a REGISTER or INVITE message, which means that the receiving SIP server can see this information.

SIP also defines *server network elements*. Although two SIP endpoints can communicate without any intervening SIP infrastructure, which is why the protocol is described as peer-to-peer, this approach is impractical for a public service. There are various implementations that can act as SIP servers:

RFC 3261 defines these server elements:

- **Proxy, Proxy Server**: An intermediary entity that acts as both a server and a client for the purpose of making requests on behalf of other clients. A proxy server primarily plays the role of routing, which means its job is to ensure that a request is sent to another entity "closer" to the targeted user. Proxies are also useful for enforcing policy (for example, making sure a user is allowed to make a call). A proxy interprets, and, if necessary, rewrites specific parts of a request message before forwarding it.
- A **registrar**: is a server that accepts REGISTER requests and places the information it receives in those requests into the location service for the domain it handles.
- A **redirect server**: is a user agent server that generates 3xx responses to requests it receives, directing the client to contact an alternate set of URIs.The redirect server allows SIP Proxy Servers to direct SIP session invitations to external domains.

The same RFC specifies: "It is an important concept that the distinction between types of SIP servers is logical, not physical."

Other SIP related network elements are

> **Session border controllers (SBC)**, they serve as "man in the middle" between UA and SIP server, see the article SBC for a detailed description.
> Various types of gateways at the edge between a SIP network and other networks (as a phone network)

- ***SIP messages***

There are two types of SIP messages: *Requests*—sent from the client to the server. *Responses*—sent from the server to the client.

REQUESTS

INVITE - Initiates a call, changes call parameters (re-INVITE).

ACK - Confirms a final response for INVITE.

BYE - Terminates a call.

CANCEL - Cancels searches and "ringing".

OPTIONS - Queries the capabilities of the other side.

REGISTER - Registers with the Location Service.

INFO - Sends mid-session information that does not modify the session state.

RESPONSES

Response messages contain numeric response codes. The SIP response code set is partly based on HTTP response codes. There are two types of responses and six classes:

RESPONSE TYPES:

- **Provisional** (1xx class)—provisional responses are used by the server to indicate progress, but they do not terminate SIP transactions

- **Final** (2xx, 3xx, 4xx, 5xx, 6xx classes)—final responses terminate SIP transactions.

CLASSES:

- 1xx = provisional, searching, ringing, queuing etc. Examples: 100- Trying, 180 – Ringing.
- 2xx = success. Example: 200 – OK.
- 3xx = redirection, forwarding. Example: 300 – Multiple choices.
- 4xx = request failure (client mistakes). Examples: 400 – Bad request, 401 – Unauthorized, 402 – Payment required (!)
- 5xx = server failures. Examples: 500 – Server internal error, 502 – Bad gateway.
- 6xx = global failure (busy, refusal, not available anywhere) – Examples: 600 – Busy everywhere, 603 – Decline.

The *Warning* field of the response also can include important information about the source of errors, like protocol errors (Incompatible network protocol, Attribute not understood), base network service errors (Multicast not available, Unicast not available), QoS problems (Insufficient Bandwidth) and other errors.

The SIP functioning is almost similar like the HTTP protocol request/response model. The begin of the every action is the request which cause an execution of a procedure. And the response informs the result.

MESSAGE PARTS

- **Start line:** Every SIP message begins with a Start Line. The Start Line conveys the message type (method type in requests, and response code in responses) and the protocol version. The Start Line may be either a Request-line (requests) or a Status-line (responses), as follows: The Request-line includes a Request URI, which indicates the user or service to which this request is being addressed. This address can be re-written by proxies. The Status-line holds the numeric Status-code and its associated textual phrase.
- **Headers:** SIP header fields are used to convey message attributes and to modify message meaning. They are similar in syntax and semantics to HTTP header fields (in fact some headers are borrowed from HTTP) and thus always take the format <name>:<value>.
  Headers can span multiple lines. Some SIP headers such as Via, Contact, Route and Request-Route can appear multiple times in a message or, alternatively, can take multiple comma-separated values in a single header occurrence.
- **Body (content)**: A message Body is used to describe the session to be initiated (for example, in a multimedia session this may include audio and video codec types, sampling rates etc.), or alternatively it may be used to contain opaque textual or binary data of any type which relates in some way to the session. Message bodies can appear both in request and in response messages. SIP makes a clear distinction between signaling information, conveyed in the SIP Start Line and headers, and the session description information, which is outside the scope of SIP.
  Possible body types include: Session Description Protocol (SDP), Multipurpose Internet Mail Extensions (MIME), Others—to be defined in the IETF and in specific implementations.

An example of the SIP request/response model:

```
                      proxy1         proxy2

   Pityu's                                       Mixer's
  SoftPhone                                     SIP Phone
     |                  |               |                  |
     | INVITE F1        |               |                  |
     |----------------->| INVITE F2     |                  |
     | 100 Trying F3    |-------------->| INVITE F4        |
     |<-----------------| 100 Trying F5 |----------------->|
     |                  |<------------- | 180 Ringing F6   |
     |                  | 180 Ringing F7|<---------------- |
     | 180 Ringing F8   |<------------- | 200 OK F9        |
     |<---------------- | 200 OK F10    |<---------------- |
     | 200 OK F11       |<------------- |                  |
     |<---------------- |               |                  |
     |                 ACK F12                             |
     |---------------------------------------------------->|
     |                 Media Session                       |
     |<===================================================>|
     |                 BYE F13                             |
     |<----------------------------------------------------|
     |                 200 OK F14                          |
     |---------------------------------------------------->|
     |                                                     |
```

Figure 5 An SIP calling set-up

- **<u>An calling set-up</u>**

First is an *INVITE* message. The message specification is below. Parameters are:

```
INVITE sip:mixer@152.66.244.188 SIP/2.0
From: <sip:pityu@152.66.244.10>
To: <sip:mixer@152.66.244.188>
Via: SIP/2.0/UDP 152.66.244.10
Contact: <sip:pityu@152.66.244.10>
Call-ID: 6585CF47-7D22-4302-AD1D-E3F1BCC0E584
CSeq: 1 INVITE
Content-Type: application/sdp
Content-Length: 117
Max-Forwards: 70

v=0
o=- 4 0 IN IP4 152.66.244.10
s=-
t=0 0
c=IN IP4 152.66.244.10
m=audio 8500 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

The first line of the INVITE message is the procedure name which will be executed. The called address (To) is only informative information. Here is also the version of the used protocol (Via).

Other messages for the successful calling set-up:

Trying message. Sent by the Proxy or UA, if it had received an INVITE message:

```
SIP/2.0 100 Trying
From: <sip: pityu@152.66.244.10>
To: <sip: mixer@152.66.244.188>
Via: SIP/2.0/UDP 152.66.244.10
Contact: <sip: pityu@152.66.244.10>
Call-ID: 6585CF47-7D22-4302-AD1D-E3F1BCC0E584
CSeq: 1 INVITE
Content-Type: application/sdp
Content-Length: 0
```

Ringing message. The called client's UA send it if it wants to accept the calling and the ringing is turn-on.

```
SIP/2.0 180 Ringing
From: <sip: pityu@152.66.244.10>
To: <sip:mixer@152.66.244.188>
Via: SIP/2.0/UDP 152.66.244.188
Contact: <sip:pityu@152.66.244.10>
Call-ID: 6585CF47-7D22-4302-AD1D-E3F1BCC0E584
CSeq: 1 INVITE
Content-Type: application/sdp
Content-Length: 0
```

OK message. The called client responds to the caller and the calling is set-up. The receiver SDP datapacket is also attached here.

```
SIP/2.0 200 OK
From: <sip: pityu@152.66.244.10>
To: <sip: mixer@152.66.244.188>
Via: SIP/2.0/UDP 152.66.244.20
Contact: <sip: mixer@152.66.244.188>
Call-ID: 6585CF47-7D22-4302-AD1D-E3F1BCC0E584
CSeq: 1 INVITE
Content-Type: application/sdp
Content-Length: 134

v=0
o=- 1 0 IN IP4 152.66.244.188
s=Answering the call
t=0 0
c=IN IP4 152.66.244.188
m=audio 8500 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

ACK message. The caller UA acknowledges the session set-up, and the data-stream is starting to flow (RTP):

```
ACK sip:mixer@152.66.244.10 SIP/2.0
From: <sip:pityu@152.66.244.20>
To: <sip:mixer@152.66.244.10>
Via: SIP/2.0/UDP 152.66.244.188
Call-ID: 6585CF47-7D22-4302-AD1D-E3F1BCC0E584
CSeq: 1 ACK
Content-Length: 0
Max-Forwards: 70
```

The session termination by the **BYE** message. This mesage can be sent by the caller or the called client. The CSEQ is incremented.

```
BYE sip:mixer@152.66.244.188 SIP/2.0
From: <sip:pityu@152.66.244.10>
To: <sip:mixer@152.66.244.188>
Via: SIP/2.0/UDP 152.66.244.188
Call-ID: 6585CF47-7D22-4302-AD1D-E3F1BCC0E584
CSeq: 2 BYE
Content-Type: application/sdp
Max-Forwards: 70
```

OK message: the session termination acknowledging.

```
SIP/2.0 200 OK
From: <sip:pityu@152.66.244.10>
To: <sip:mixer@152.66.244.188>
Via: SIP/2.0/UDP 152.66.244.10
Call-ID: 6585CF47-7D22-4302-AD1D-E3F1BCC0E584
CSeq: 2 BYE
Content-Type: application/sdp
Content-Length: 0
```

5 Programs and tools used during measurement

During the measurement your task will be to connect two SIP based VOIP tool (one SIP telephone and one softphone program executed on a PC) and one destination oriented, freely parameterized and controlled VOIP-able software, and inspection of their package-level traffic, and also study the speechcoding process usable during the session (connection). During building this session (connection), for easier distinctness, we won't face with the special services of SIP, therefore there is no proxy, redirect, registrar, etc. servers on the route, furthermore we won't use DNS based name - resolve function. Therefore we will quote the tools with an IP address within the SIP messages or the used softphone program.

- ***Tools / software used during measurement:***

a) An IPTeL destination – oriented software will help the exercise, which can process RTP packages, so we set the caller and also the called partner to expect an RTP package and we send a voicemail coded to the port. The program will decode this, and send it to the tool connected with the PC voicecard, and reverse. Besides, the program can compile SIP messages (in other words can compile optional messages above UDP) and if asked can send the message to the 5060 port of the given address and displays the answer message.
b) During the measurement you will need a freely usable softphone program, which we can build regular SIP based voice connections with, in our case this is a software called SJphone, freely downloadable and usable from Internet. Using this is evident, so we won't explain it.
c) An additional tool is a Linksys SPA921 type SIP-able VoIP telephone, user guide can be seen at the measurement site.
d) For analyzing the traffic we use a program called Wireshark, which you know from previous studies.

- ***Functions and operation of IPtel program***

IPtel program is based on open sourced WinRTP stack, where we extended its functionality, and made it parameterizable. It supported G.711 coding in base building, and we completed it with codings as follows:

1. G.729 Annex A
2. GSM
3. AMR
4. G.723.1

IPtel client has the following characteristics:

1. Adjustable destination station IP address and port address
2. Various coding methods and coding rate
3. Voice Activity Detection (VAD) – speech gaps detection
4. Automatic Gain Control (AGC) – automatic setting of microphone sensibility
5. Changing frame size
6. Changing volume and microphone gain
7. Display parameters of incoming traffic

At starting the program will load the last used setting. At first start default coding is G. 711 u-law, IP address is 152.66.244.29 and the port number is 8500. Unfortunately the conception of the basic program does not allow the automatic detection of the coding algorithm, so it uses the same algorithm for decoding and coding. Therefore you have to watch out when setting the coding, and if e.g. the sender somehow changes the coding algorithm, it is rational to switch off sending, and restart after setting the recipient and the sender, or if not, you have to accept that the communication works only one way.

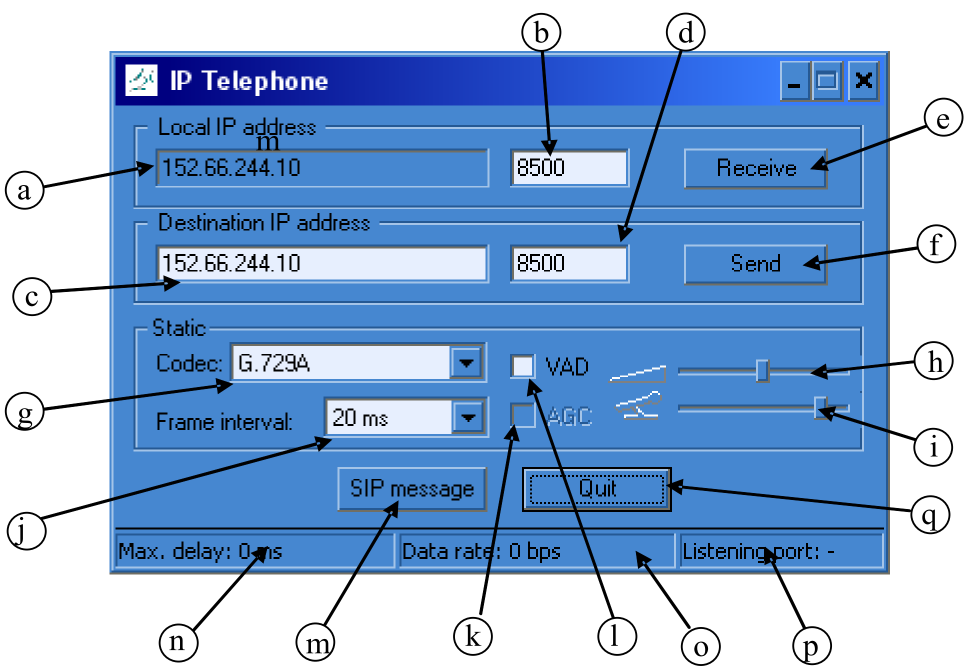


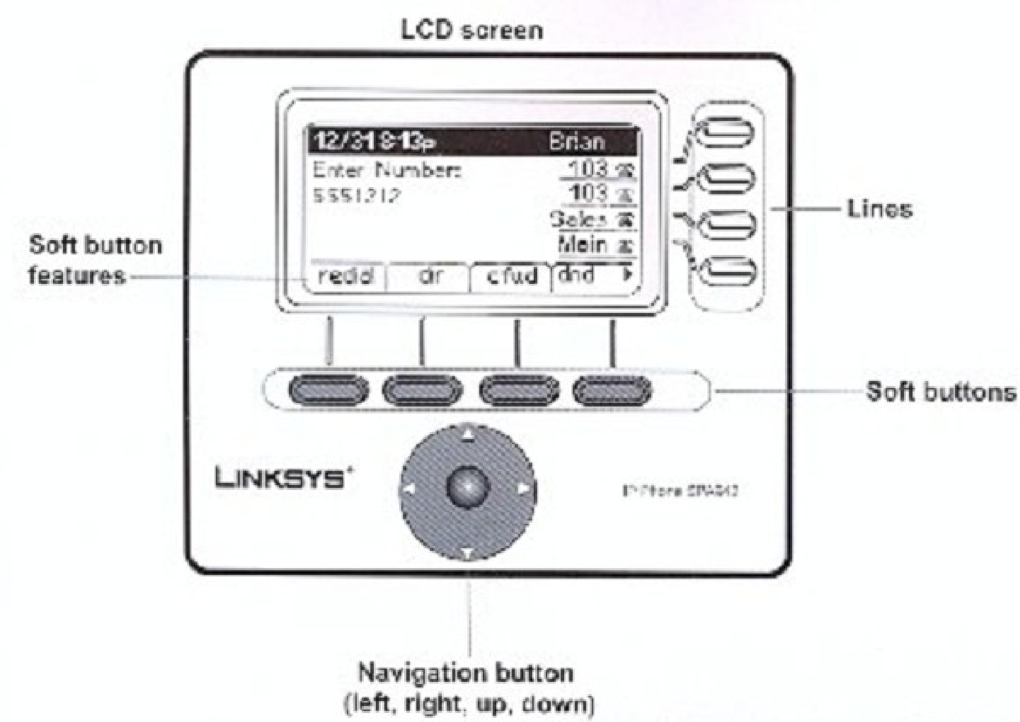
Figure 6 The client user-interface

Elements of the user-interface are:

- IP address of local station
- Port number of local monitoring
- IP address of destination. This is the address of the server if we try in central mode, or another IPtel client IP address.
- Port number of destination. In case of the server it can be anything, otherwise the monitoring port of the other client is 8500.
- Switching on/off the reception
- Switching on/off the sending
- Choosing the actual coding
- Volume adjusting
- Microphone sensibility adjusting
- Size of the actual frame in ms
- AGC algorithm switch button. AGC can operate only after switching on VAD
- VAD switch button
- Sending SIP message
- Maximum of arriving traffic frame-delay
- Bandwidth of arriving traffic
- Local monitoring port
- Exit

- *__The Linksys SPA921 VoIP phone functions and operation__*

Figure 7 The Linksys SPA921 phone user-interface



LCD Screen Overview

- Top line displays date, time, phone number
- main area displays call information
- bottom lines display soft button options
- right side displays extension numbers, if available.
- The SPA962 has a 320 x 240 pixel color display

Soft Button Features

Shows available features.
As shown above, by pressing the soft button below the feature, you could:

redial: view redial list
dir: view directory options
cfwd: forward your calls
dnd: choose do not disturb.

In this display, more choices are available by pressing the right navigation button.

Soft Buttons

Press to activate a soft button feature.

Navigation button

Use to move up, down, left or right through soft button features.

Lines

(SPA 941, SPA942 and SPA962 only)
Use to access additional extensions.
The SPA941 and SPA942 have four lines, the SPA962 has six lines.

Figure 8 The Linksys SPA921 phone screen

1. For the following information ask the system administrator

- IP address of SIP proxy (e.g. fwd pulver .com)
- IP address of outgoing SIP proxy
- User identifier and password
- OHCP access or static IP (see 13. about static IP setting)
- Sub network mask (if needed for the static IP network

- IP address of gateway (if necessary for static IP network)

2. Inquire the IP address of SPA 921 telephone: press MENU button on the calling table of the telephone. With the help of arrows up and down go to menu ”Network” and press “Select”. IP address will occur in “Current IP” field. Write down the IP address, because you will need it later.
3. write IP address to the web browser in your computer joint to the sub network, and press Enter. The browser will show the menu of SPA 921 telephone.
4. Click “Admin login” with the mouse, and then “Advanced” link on right of the sheet.
5. Click on “Ext 1” sheet. Go to “Proxy and registration”. Write the IP or URL address of SIP Proxy in “Proxy”, and press Enter.
6. Write the IP or URL address of outgoing proxy in “Outgoing proxy” and press enter
7. Go to “Subscriber information”. Write user identifier in “User ID” and press “Enter”.
8. Write “Password” and press “Enter”
9. Press “Submit All Changes” at the bottom of the sheet to save changes.
10. If you use OHCP, the setting has finished. In other case go to 14.
11. Test call: pick up the phone. If registration was successful, you will hear dial tone. Dial a valid phone number. When finishing the call put down the receiver.
12. In case the SPA-921 does not use a OHCP, choose “System” sheet.
    a) Set OHCP on “No”
    b) Write the IP address in “Static IP”
    c) Write the address of sub network mask in “NetMask”
    d) Write gateway IP address in “Gateway”
    e) Write the address of ONS server (optional)
    f) Press button “Submit All Changes” at the bottom of the page, to save changes
    g) Do a test call as written in 11.

13. If you use an SPA 921 NAT device (e.g. router) use the following settings
    a) Roll down to “NAT settings” on page “Ext 1”
    b) “NAT Mapping Enable” set “Yes”
    c) “NAT Keep Alive Enable “ set Yes”
    d) Then press “Submit All Changes” at the bottom of the page to save changes
    e) Do a test call as written in 11.

Attention! Changes will be saved only in case you press “Submit All Changes” button at the bottom of the page! More info on the settings is at www.Linksys.com/support .

Functions and Set up of Physical characteristics of SPA 921:

1. 4 programmable buttons: Programmable buttons can be used for access of distinctive characteristics or showing options
2. Navigation button: you can move up, down, left, right
3. Voicemail service button: user can automatically call the voice letter system. The button will dial the user's postbox immediately
4. Holding button: The button will hold the call, the two parties will not hear each other
5. Info button: The user can reach the options of the menu. These will help the user in settings and look into the parameters
6. Mute button: you can switch off the microphone. The red LED will show that the appliance is muted
7. Earphone button: this will allow using earphone. A green LED will show if the headset is active.
8. Volumes adjust. With this button the user can adjust the volume of ringing, handset, earphone and the speaker.
9. Speakerphone button. You can change between switching on and off the speakerphone. Green LED in on when it is on.
10. Calling button. The indicator is red when there is a call
11. Display: It will show the state of the call, dialing, address list and menu.
12. Numeric keyboard: The user can indicate phone numbers and menu options.

Recommanded literature

http://www.cs.columbia.edu/sip/ - SIP website

www.ietf.org - The RFC-s website.