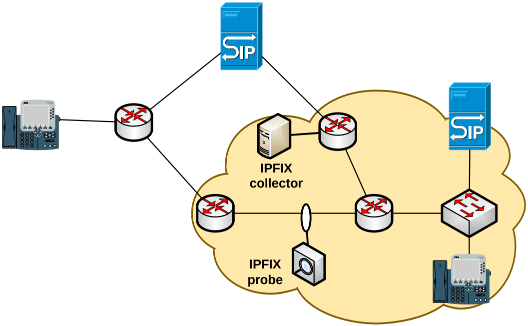
On-line Monitoring Of VoIP Quality Using IPFIX

Petr Matoušek, Martin Kmeť and Martin Basel

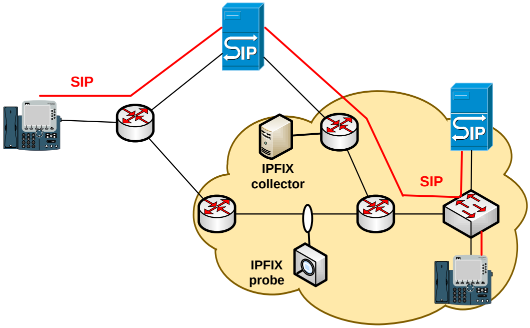
Brno University of Technology, Czech Republic *matousp@fit.vutbr.cz, ikmet@fit.vutbr.cz, xbasel02@stud.fit.vutbr.cz*

Knowledge in Telecommunication Technologies and Optics 2014
Malenovice, September 3^{rd}–5^{th}, 2014

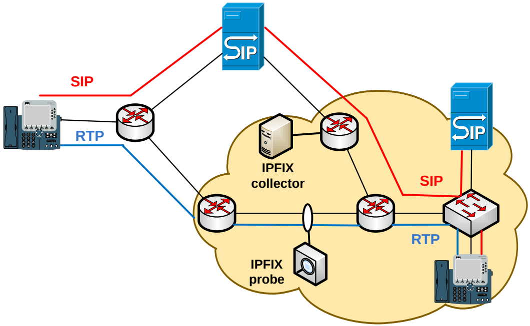
On-line Network Monitoring



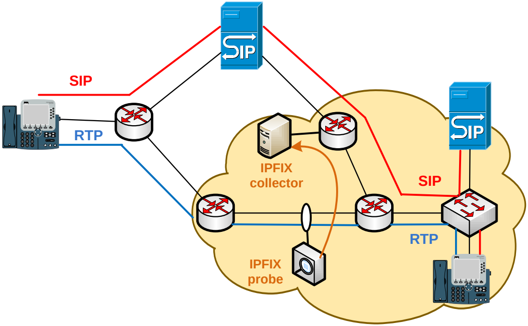
On-line Network Monitoring



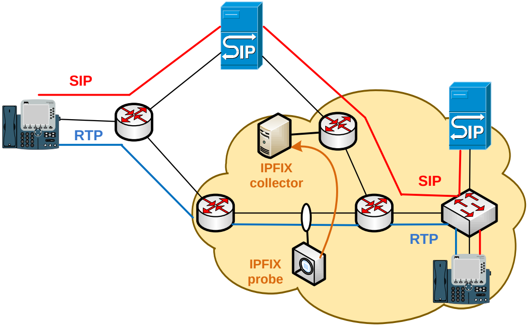
On-line Network Monitoring



On-line Network Monitoring



On-line Network Monitoring



Requirements for VoIP Quality Monitoring

- Assessment of VoIP quality of users' calls
- Long-term monitoring of the quality with history
- Integration with common network monitoring systems
- End communication point outside the monitoring network

Talk overview

1. Motivation
   - On-line VoIP Quality Monitoring
   - Current Methods
2. Computational Model
   - Simplified E-Model
   - Implementation Issues
3. Results
   - IPFIX Extension
   - Result validation
4. Conclusion
   - Future Work

Existing methods for evaluating quality of VoIP calls

Subjective methods

- Listening-opinion tests (ITU-T P.800, ACR)

Existing methods for evaluating quality of VoIP calls

Subjective methods

- Listening-opinion tests (ITU-T P.800, ACR)

Objective methods

- Intrusive measurement
  - Comparing the reference and the degraded speech signal (PESQ)

Existing methods for evaluating quality of VoIP calls

Subjective methods

- Listening-opinion tests (ITU-T P.800, ACR)

Objective methods

- Intrusive measurement
  - Comparing the reference and the degraded speech signal (PESQ)
- Non-intrusive measurement
  - Passive method that does not require the original signal (E-model)

Existing methods for evaluating quality of VoIP calls

Subjective methods

- Listening-opinion tests (ITU-T P.800, ACR)

Objective methods

- Intrusive measurement
  - Comparing the reference and the degraded speech signal (PESQ)
- Non-intrusive measurement
  - Passive method that does not require the original signal (E-model)

- On-line monitoring requires non-intrusive passive approach.

E-Model (ITU-T G.107)

E-Model (ITU-T G.107)

- Considers all possible impairments for an end-to-end speech transmission.
- Factors such as SNR (R_0), quantization distortation (I_s), delay (I_d), codecs, packet loss, jitter (I_{e-eff})

E-Model (ITU-T G.107)

- Considers all possible impairments for an end-to-end speech transmission.
- Factors such as SNR (R_0), quantization distortation (I_s), delay (I_d), codecs, packet loss, jitter (I_{e-eff})

Original E-Model (ITU-T G.107, 12/2011)

$$R = R_o - I_s - I_d - I_{e-eff} + A$$

E-Model (ITU-T G.107)

- Considers all possible impairments for an end-to-end speech transmission.
- Factors such as SNR (R_0), quantization distortation (I_s), delay (I_d), codecs, packet loss, jitter (I_{e-eff})

Original E-Model (ITU-T G.107, 12/2011)

$$R = R_o - I_s - I_d - I_{e-eff} + A$$

- What if we replace factors, that we cannot measure, by default values and focused only on network parameters?

E-Model (ITU-T G.107)

- Considers all possible impairments for an end-to-end speech transmission.
- Factors such as SNR (R_0), quantization distortation (I_s), delay (I_d), codecs, packet loss, jitter (I_{e-eff})

Original E-Model (ITU-T G.107, 12/2011)

$$R = R_o - I_s - I_d - I_{e-eff} + A$$

- What if we replace factors, that we cannot measure, by default values and focused only on network parameters?

Simplified E-Model (using default values for R_o, I_s, and A)

$$R = 93.2 - I_d - I_{e-eff}$$ [i]

[i] L. Sun et al.: *Guide to Voice and Video over IP*, Springer, 2013, p. 147

Simplified E-Model used in this work

Computation of the Simplified E-Model

$$R = 93.2 - I_d - I_{e-eff}$$

Simplified E-Model used in this work

Computation of the Simplified E-Model

$$R = 93.2 - I_d - I_{e-eff}$$

$$I_d = \begin{cases} 0,0267 \cdot d & d < 175\ ms \\ 0,1194 \cdot d - 15,876 & 170\ ms \leq d \leq 400\ ms \end{cases}$$ [ii]

[ii] M.Vozňák: *Non-intrusive Speech Quality Assessment in Simplified E-Model*, WSEAS, vol. 11, 2012

Simplified E-Model used in this work

Computation of the Simplified E-Model

$$R = 93.2 - I_d - I_{e-eff}$$

$$I_d = \begin{cases} 0,0267 \cdot d & d < 175\ ms \\ 0,1194 \cdot d - 15,876 & 170\ ms \leq d \leq 400\ ms \end{cases}$$ [ii]

$$I_{e-eff} = I_e + (95 - I_e) \cdot \frac{P_{plef}}{P_{plef} + B_{pl}}$$ [iii]

[ii] M.Vozňák: *Non-intrusive Speech Quality Assessment in Simplified E-Model*, WSEAS, vol. 11, 2012

[iii] ITU-T G.107: *The E-model, a computational model for use in transmission planning*, 2001, p. 7

Simplified E-Model used in this work

Computation of the Simplified E-Model

$$R = 93.2 - I_d - I_{e-eff}$$

$$I_d = \begin{cases} 0,0267 \cdot d & d < 175 \ ms \\ 0,1194 \cdot d - 15,876 & 170 \ ms \leq d \leq 400 \ ms \end{cases}$$ [ii]

$$I_{e-eff} = I_e + (95 - I_e) \cdot \frac{P_{plef}}{P_{plef} + B_{pl}}$$ [iii]

$$P_{plef} = P_{pl} + P_{jitter} - P_{pl} \cdot P_{jitter}$$ [iv]

[ii] M.Vozňák: *Non-intrusive Speech Quality Assessment in Simplified E-Model*, WSEAS, vol. 11, 2012

[iii] ITU-T G.107: *The E-model, a computational model for use in transmission planning*, 2001, p. 7

[iv] M. Halas, et al.: *Computationally efficient E-model improvement of MOS estimate*, TSP, 2012, pp. 86–90

Simplified E-Model used in this work

Computation of the Simplified E-Model

$$R = 93.2 - I_d - I_{e-eff}$$

$$I_d = \begin{cases} 0,0267 \cdot d & d < 175\ ms \\ 0,1194 \cdot d - 15,876 & 170\ ms \leq d \leq 400\ ms \end{cases}$$ [ii]

$$I_{e-eff} = I_e + (95 - I_e) \cdot \frac{P_{plef}}{P_{plef} + B_{pl}}$$ [iii]

$$P_{plef} = P_{pl} + P_{jitter} - P_{pl} \cdot P_{jitter}$$ [iv]

- Depends on a codec (B_{pl}), network delay (d), packet loss (P_{pl}), and network jitter (P_{jitter}).

[ii] M.Vozňák: *Non-intrusive Speech Quality Assessment in Simplified E-Model*, WSEAS, vol. 11, 2012

[iii] ITU-T G.107: *The E-model, a computational model for use in transmission planning*, 2001, p. 7

[iv] M. Halas, et al.: *Computationally efficient E-model improvement of MOS estimate*, TSP, 2012, pp. 86–90

Simplified E-Model used in this work

Computation of the Simplified E-Model

$$R = 93.2 - I_d - I_{e-eff}$$

$$I_d = \begin{cases} 0,0267 \cdot d & d < 175\ ms \\ 0,1194 \cdot d - 15,876 & 170\ ms \leq d \leq 400\ ms \end{cases}$$ [ii]

$$I_{e-eff} = I_e + (95 - I_e) \cdot \frac{P_{plef}}{P_{plef} + B_{pl}}$$ [iii]

$$P_{plef} = P_{pl} + P_{jitter} - P_{pl} \cdot P_{jitter}$$ [iv]

- Depends on a codec (B_{pl}), network delay (d), packet loss (P_{pl}), and network jitter (P_{jitter}).
- Values are computed on-the-fly using packet headers.

[ii] M.Vozňák: *Non-intrusive Speech Quality Assessment in Simplified E-Model*, WSEAS, vol. 11, 2012

[iii] ITU-T G.107: *The E-model, a computational model for use in transmission planning*, 2001, p. 7

[iv] M. Halas, et al.: *Computationally efficient E-model improvement of MOS estimate*, TSP, 2012, pp. 86–90

Phases of monitoring VoIP calls

1. Filtration of calls from network traffic

Using SIP signalling or directly from RTP packets

RTP packets and voice codecs detected based on packet features

Phases of monitoring VoIP calls

1. Filtration of calls from network traffic

Using SIP signalling or directly from RTP packets
RTP packets and voice codecs detected based on packet features

2. Monitoring of RTP/RTCP flows

Monitored factors: Jitter, packet loss, one-way delay, type of codec

Phases of monitoring VoIP calls

1. Filtration of calls from network traffic

Using SIP signalling or directly from RTP packets
RTP packets and voice codecs detected based on packet features

2. Monitoring of RTP/RTCP flows

Monitored factors: Jitter, packet loss, one-way delay, type of codec

3. Voice Quality Assessment

Computing R-factor and MOS using simplified E-model.

Phases of monitoring VoIP calls

1. Filtration of calls from network traffic

Using SIP signalling or directly from RTP packets
RTP packets and voice codecs detected based on packet features

2. Monitoring of RTP/RTCP flows

Monitored factors: Jitter, packet loss, one-way delay, type of codec

3. Voice Quality Assessment

Computing R-factor and MOS using simplified E-model.

4. Export of Flows

IPFIX records extended by packet loss, jitter, delay, R-factor, and MOS

RTP Detection and Codecs Classification

RTP Detection and Codecs Classification

- RTP streams detected using specific values in UDP/RTP header (RFC 3550/A)
  - UDP only, ports > 1024, min. header length, RTP version 2, RTP PT

RTP Detection and Codecs Classification

- RTP streams detected using specific values in UDP/RTP header (RFC 3550/A)
  - UDP only, ports > 1024, min. header length, RTP version 2, RTP PT
- Codecs classified using a specific feature set[v]

[v] M. Kmeť: *Analysis and Detection of Multimedia Types in RTP Traffic*, MSc. Thesis, BUT, Brno, 2014

RTP Detection and Codecs Classification

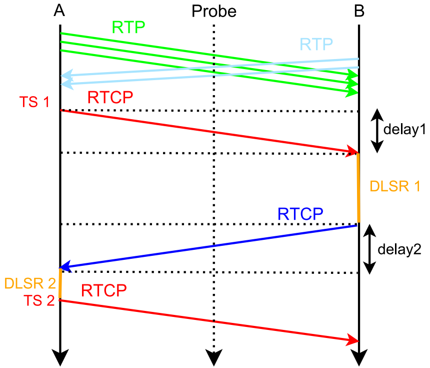
- RTP streams detected using specific values in UDP/RTP header (RFC 3550/A)
  - UDP only, ports > 1024, min. header length, RTP version 2, RTP PT
- Codecs classified using a specific feature set[v]

Codec	Payload Type	Δ time ($TS_{i+1} - TS_i$)	Payload Size	Δ ratio
G.711 μ-law	0	160	160	1:1
G.711 A-law	8	160	160	1:1
Speex8	dyn	160	20	8:1
Speex16	dyn	320	52	80:13
GSM	3	160	33	160:33
G.722	9	160	160	1:1
G.722.1	dyn	320	60	16:3
G.723.1-5k	4	240	20	12:1
G.723.1-6k	4	240	24	10:1
G.726-16	dyn	80/240	20/60	4:1
G.726-24	dyn	80/240	30/90	8:3
G.726-32	dyn	80/240	40/120	2:1
G.726-40	dyn	80/240	50/150	8:5
G.729	18	160	20	8:1
G.729a	18	160	20	8:1
G.729b	18	160*	20*	var.
AMR-WB	dyn	320	62	160:31
AMR-12k	dyn	160	33	160:33
Silk8	dyn	320	var.	var.
Silk16	dyn	640	var.	var.

[v] M. Kmeť: *Analysis and Detection of Multimedia Types in RTP Traffic*, MSc. Thesis, BUT, Brno, 2014

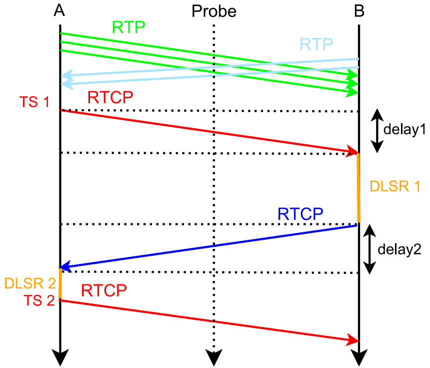
Computing One-Way Delay

Computing One-Way Delay



- Using RTCP Packets: $RTD_{12} = TS_2 - DLSR_2 - DLSR_1 - TS_1$
  - Average RTD delay over all RTCP packets considered.

Computing One-Way Delay



- Using RTCP Packets: $RTD_{12} = TS_2 - DLSR_2 - DLSR_1 - TS_1$
  - Average RTD delay over all RTCP packets considered.
- Using RTP packets only: $d_i = \frac{d_{i-1}.(i-1)+J_i}{i}, d_1 = 0$ [vi]
  - Delay d computed iteratively over subsequent packets.

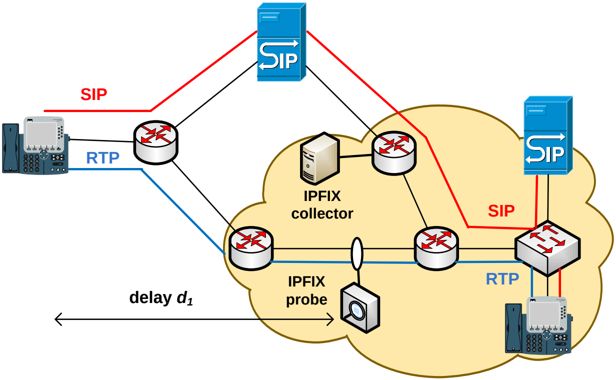
[vi] see http://www.voipmonitor.org

Location of probe matters

- Which values are more accurate? RTP or RTCP?

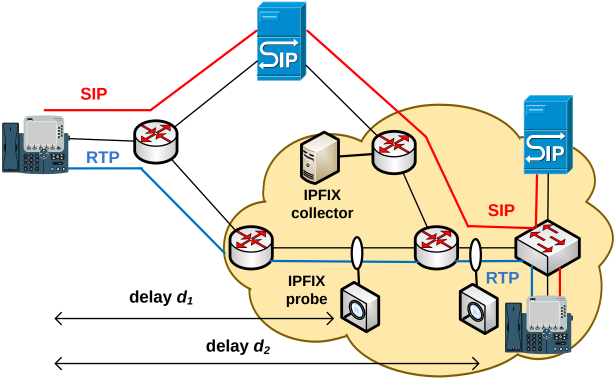
Location of probe matters

- Which values are more accurate? RTP or RTCP?



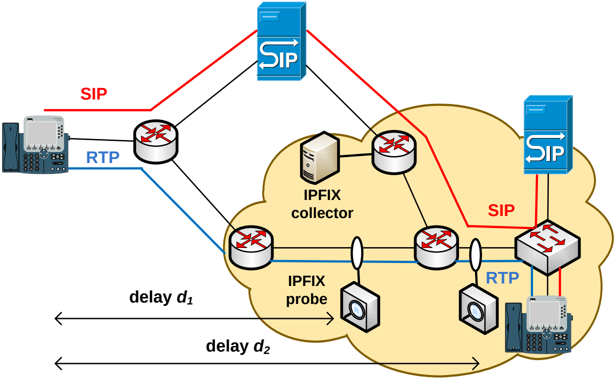
Location of probe matters

- Which values are more accurate? RTP or RTCP?



Location of probe matters

- Which values are more accurate? RTP or RTCP?



- RTCP delay computed at end-points ⇒ more accurate

Implementation results

IPFIX extension: Example

- Packet Loss (0), Jitter (0.201), Delay (0), R-Factor (93.2), MOS (4.409), Quality (Excellent), Flow Type (RTP)

Implementation results

IPFIX extension: Example

- Packet Loss (0), Jitter (0.201), Delay (0), R-Factor (93.2), MOS (4.409), Quality (Excellent), Flow Type (RTP)

```
⊟ Cisco NetFlow/IPFIX
    Version: 10
    Length: 380
  ⊞ Timestamp: Apr  2, 2014 12:35:26.000000000 CEST
    FlowSequence: 0
    Observation Domain Id: 0
  ⊟ Set 1
      FlowSet Id: (Data) (258)
      FlowSet Length: 364
    ⊟ Flow 1
        Octets: 178600
        Packets: 893
      ⊞ [Duration: 17.839000000 seconds]
        InputInt: 0
        OutputInt: 0
        IPVersion: 04
        SrcAddr: 91.221.212.167 (91.221.212.167)
        DstAddr: 192.168.1.4 (192.168.1.4)
        IP ToS: 0x00
        Protocol: 17
        SrcPort: 26456
        DstPort: 7078
        Enterprise Private entry: (VUT BRNO, faculty of EE and CS) Type 721: Value (hex bytes): 00
        Enterprise Private entry: (VUT BRNO, faculty of EE and CS) Type 722: Value (hex bytes): 3f 61 fb 9a
        Enterprise Private entry: (VUT BRNO, faculty of EE and CS) Type 723: Value (hex bytes): 00 00 00 10
        Enterprise Private entry: (VUT BRNO, faculty of EE and CS) Type 724: Value (hex bytes): 42 b9 03 00
        Enterprise Private entry: (VUT BRNO, faculty of EE and CS) Type 725: Value (hex bytes): 40 8c a7 83
      ⊞ [Enterprise Private entry: (VUT BRNO, faculty of EE and CS) Type 726: Value (hex bytes): 47 6f 6f 64
      ⊞ [Enterprise Private entry: (VUT BRNO, faculty of EE and CS) Type 727: Value (hex bytes): 52 54 50 00
    ⊞ Flow 2
    ⊞ Flow 3
```

Comparison with other software

- VoIP metrics computed by Wireshark (W), PacketScan (PS), VoIPmonitor (VPM) and IPFIX plugin (IPF)

Metric	W	PS	VPM	IPF
RTP Jitter (ms)	8.10	7.00	8.00	7.91
RTP Loss (%)	0.00	0.00	0.00	0.00
RTP Delay (ms)	–	0.00	–	0.00
R-factor	–	93.0	–	92.99
MOS	–	4.20	4.50	4.41
RTCP Jitter (ms)	–	0.00	76.10	76.06
RTCP Loss (%)	–	0.00	0.00	0.00
RTCP Delay (ms)	–	9.23	–	15.00
R-factor	–	–	–	90.09
MOS	–	–	–	4.34

Future research

Future research

Qualitative testing

- Influence of different codecs on computation
- Advanced methods how to measure one-way delay

Future research

Qualitative testing

- Influence of different codecs on computation
- Advanced methods how to measure one-way delay

Monitoring in production networks

Future research

Qualitative testing

- Influence of different codecs on computation
- Advanced methods how to measure one-way delay

Monitoring in production networks

- Long-term monitoring of VoIP calls quality using IPFIX
  - Correlation among different quality parameters over time period

Future research

Qualitative testing

- Influence of different codecs on computation
- Advanced methods how to measure one-way delay

Monitoring in production networks

- Long-term monitoring of VoIP calls quality using IPFIX
  - Correlation among different quality parameters over time period
- Monitoring of VoIP calls quality using IPFIX probes in different nodes
  - Correlation among different quality parameters over time period

Thank you for your attention.