Optimizing AoIP links with AETA codecs

Contents

1. Introduction 1

2. Components of the quality of an AoIP link 1

2.1. Audio coding quality 1

2.2. Network quality of service (QoS) 1

2.3. Global latency 2

3. Impact of the various available settings 3

3.1. Bit rate reduction algorithm 3

3.2. Jitter buffer size 3

3.3. Redundancy 4

3.4. Multiple network paths 4

3.5. Summary table 5

4. AETA toolbox 6

4.1. Robustness improvement tools 6

4.2. Simplification tools 6

5. Use cases 8

1. Introduction

This note describes the parameters that can influence the quality of an AoIP link, and details the specific effects of the various parameters and techniques available to optimize these links.

We also describe the tools available to help or facilitate these adjustments.

As a general rule, the use of the SIP protocol and a SIP server is recommended, at least to facilitate the establishment of links. However, some of the settings or methods mentioned here are still relevant even if you use more basic modes of operation.

2. Components of the quality of an AoIP link

2.1. Audio coding quality

The performance of the audio coding used is an essential component, but not specific to IP transmissions. However, the data rate to be transmitted has an impact, and coding algorithms offering more flexibility in this respect are useful for the optimization of links.

2.2. Network quality of service (QoS)

Through an IP transmission network, the quality of service can be analyzed and measured according to various criteria.

2.2.1. Availability

Availability, often measured as a proportion of the time during which the network is operational, is obviously an important element for the implementation of links.

2.2.2. Data losses

On IP networks, transmission errors and data losses result in packet losses. One of the most common causes is network congestion, such as when a busy router drops packets without forwarding them.

Packet losses strongly affect the transmission of real-time media streams since these streams typically use the UDP protocol.

2.2.3. Latency

We are talking here about network latency: transmission delay from the sender of a packet to its destination. High latency is not in itself a quality defect, but it can limit transmission performance in some configurations.

2.2.4. Transmission jitter

In an ideal network, the above latency would be a deterministic constant. In fact this is not the case in a real situation; the delay varies from packet to packet, due to variations in processing by routers along the way, or even due to route changes. A regular flow at the source therefore arrives in a more or less irregular pattern, a phenomenon called "jitter".

Jitter is a critical defect for a media stream, that requires smooth and uninterrupted delivery. To compensate for it, a stream receiver uses a reception buffer (jitter buffer) which absorbs the variations.

2.3. Global latency

Overall latency, the delay in transmission from the audio source to the sound output at the destination, is a very critical characteristic for many real-time audio transmission applications. This latency is a sum of factors:

- Audio processing latency in the involved equipment, generally negligible in the context of an AoIP link.
- **Audio coding** latency: this is highly variable depending on the coding algorithm, and some can reach a very high latency.
- **Network** latency as described above (2.2.3).
- **Jitter buffer** latency: by nature this buffer applies a delay, longer if the buffer can absorb high jitter. Conversely, a small size buffer can only handle limited jitter.

This multiplicity of factors makes optimizing latency rather difficult.

3. Impact of the various available settings

3.1. *Bit rate reduction algorithm*

Apart from the fidelity or "transparency" of the coding algorithm, the transmission rate is a factor which has a significant impact. A low throughput flow is, as a rule, less likely to suffer from degradation due to network congestion.

In other words, in a situation of heavy traffic on the network, it is possible to seek to reduce disturbances by going for a lower bit rate. This is not possible for all coding algorithms, for example G711 / G722 coding have a fixed bit rate.

Furthermore, more efficient coding in terms of compression rate is advantageous, allowing better audio quality at a given bit rate.

Another major impact of the choice of coding is the coding latency, which varies greatly depending on the algorithm. Low latency algorithms are preferred when this criterion is critical for the intended application.

The Opus algorithm is particularly interesting on both levels, since at the same time it has very good compression efficiency, a very flexible bit rate, and low latency. Lastly, a big advantage is its ability to change bitrate on the fly, without interrupting or disturbing the transmission.

3.2. *Jitter buffer size*

A larger or "deeper" buffer absorbs strong fluctuations, while a small size buffer may be insufficient for this. However, a large buffer also implies a higher latency.

Adjusting the buffer size is therefore a compromise between two conflicting needs:

- For maximum stability when experiencing big network disturbances, it is best to select a large buffer size.
- But one has to reduce this size if one wants to limit the latency.

On an AETA codec, this choice is behind the "Network quality" setting proposed for each IP transmission interface: poor network quality imposes a larger buffer at the cost of higher latency, while a good quality allows for a smaller buffer and low latency.

Important notes:

- This setting on a device affects the reception quality; it has no effect on the quality of the "transmission" direction towards the remote device.
- The size of the receive buffer has no direct influence on the tolerance to packet losses. On the contrary, an oversized jitter buffer can even worsen the undesirable effects of a momentary interruption of reception.

3.3. Redundancy

With UDP streaming, lost packets are not retransmitted, and, if they are numerous, cause unwanted noise and dropouts. Redundancy is a workaround against this problem, with the Forward Error Correction (FEC) system: redundant data is transmitted at the same time as the stream, and the receiver can rebuild the missing data without interaction with the source. The tolerance to losses can be greatly improved. Unfortunately, this also has drawbacks:

- The throughput is increased in proportion to the redundancy rate; this can degrade the transmission quality and therefore thwart the improvements.
- The latency is generally increased.

AETA codecs offer a 100% redundancy variant with total packet duplication; this option has the advantage of very little additional latency.

3.4. Multiple network paths

Using more than one path for the transmission of the media stream is a powerful way to increase robustness; on AETA codecs this is "Double Streaming" which allows transmitting at the same time through two network access interfaces.

The most efficient case is a "spatial diversity" variant of 100% redundancy: the same stream is transmitted simultaneously on two routes. This technique brings big advantages:

- Packets lost on one branch are almost always well received on the other, since the two routes are independent. The overall loss rate remains very low even with two poor quality paths.
- At worst, the complete loss of one of the two routes only slightly degrades the overall quality.
- The system does not overload either of the two routes, unlike the redundancy mentioned above.
- Latency is little or not affected.

Another case is the distribution of the stream on the two routes ("bonding"):

- The load is shared on the two paths, therefore with a lower speed on each, which may be better suited to a very busy network.
- On the other hand, the vulnerability to temporary or permanent transmission cutoff is increased compared to normal transmission via a single route.

3.5. *Summary table*

The table below summarizes the various levers which make it possible to act on the quality of transmission, recalling their respective strengths and weaknesses.

Action	Advantages	Disadvantages	Notes and supplements
Reduction of the bit rate	Less demand on the network, thus less risk of overloading	Reduced audio quality if the bit rate is very low Not possible for some coding algorithms	Prefer high efficiency coding algorithms: HE-AAC, Opus Opus allows to act in real time
Increased jitter buffer size (More pessimistic "Network quality" setting)	Better jitter tolerance	Higher latency	Effective only in the *receiving* direction
Packet duplication	Improved tolerance to losses	Doubled network bit rate: risk of worsening the congestion on a loaded network	Combine with a reduced bit rate to compensate for the increase. Effective only in the *transmitting* direction.
Double Streaming **(100% diversity)**	Very high loss tolerance, including total loss of one of the streams	Requires the use of two transmission media, relatively complex	Only with SIP
Double Streaming **("bonding" load balancing)**	Reduced load on each access used: potentially less congestion	Requires the use of two transmission media, relatively complex Increased vulnerability to network disturbance	Only with SIP

4. AETA toolbox

4.1. Robustness improvement tools

These tools make it possible to implement transmission with redundancy in order to reinforce the robustness of links subject to data losses.

4.1.1. Packet duplication

AETA codecs offer, among the AoIP parameters, an RTP transmission mode with enhanced robustness using packet duplication. When this mode is enabled, each packet is transmitted twice; with this system a lost packet has no effect because the receiver still gets the other copy of the packet. Stable links are thus obtained even with a high rate of packet losses. Of course, the counterpart is a doubled bit rate; it is therefore necessary to ensure that this speed remains compatible with the means of transmission used.

Two variants of duplication are available:

- Packet duplication without interleaving
- Packet duplication with interleaving: the second packet is shifted in time; robustness is enhanced but latency is slightly increased

You can refer to the user manuals for operation details.

4.1.2. Double Streaming

Double Streaming is based on setting up two SIP links, simultaneous but using two separate routes. On each codec, a specific transmission interface can be chosen for each of the two AoIP connections. It is thus possible to create two independent routes, the possible faults of which are not correlated. These two routes can be operated using the following two techniques.

In the case of "diversity" redundancy, the secondary link carries a copy of the audio stream from the primary link. The codec which receives the two (normally) identical streams recombines them, discarding duplicates. This technique is very effective in obtaining a strong link, resisting even the complete loss of one of the streams.

In the case of distribution or "bonding", the initial stream sent by the audio encoder is distributed over the two links. On receipt of the two streams at the other end, the data is recombined to restore the initial stream. Thus each only carries a lower rate (typically half), more likely to be transmitted correctly if the links have a limited throughput.

The implementation and settings of the "Double Streaming" tool are described in detail in an application note, available on the AETA website:

"Using the Double Streaming and Bonding features"

4.2. Simplification tools

The adjustments or changes on the coding rate or the size of the jitter buffer may look complex, and sometimes require transmission tests to iteratively search for the appropriate settings, using the quality measurements provided by the codecs.

The "Auto Bit Rate Opus" and "Auto Jitter Buffer" tools are ways to facilitate these settings. They are presented succinctly below, but an application note, available on the AETA website, describes all the details on the implementation and settings of these tools:

"How to use Auto Bit rate Opus and Auto Jitter Buffer"

4.2.1. Auto Bit Rate Opus

With adjustable bit rate encoding such as Opus, the choice of bit rate is usually a compromise between two competing needs:

- The search for the best audio quality naturally leads to prefer the highest possible bit rate;
- But a restricted speed is preferable to minimize the risk of packet loss when the network is loaded or even congested.

The automatic bit rate system will avoid making this choice *a priori*, by automatically adjusting the speed during transmission. If the quality decreases the system will reduce the bit rate to seek better robustness, conversely it will increase this bit rate if the quality is good and stable.

This system only works with Opus coding. It is not compatible with "Double Streaming".

4.2.2. Auto Jitter Buffer

Adjusting the jitter buffer size is a compromise between two competing needs:

- For maximum stability when subject to large network disturbances, it is best to select a large buffer size.
- But you have to reduce this size if you want to limit the latency.

The right setting can be tricky to make; for example it can be facilitated by a link test and observation of the available quality measurements (jitter measurement in this case).

The “Auto Jitter Buffer” adaptive buffer system will avoid making this choice *a priori*, by automatically adjusting the buffer during transmission. If the jitter increases the system will increase the size of the buffer to seek better stability, conversely it will reduce its size if the jitter turns out to be low, thus reducing the latency.

The system is active regardless of the AoIP protocol, whether SIP is used or not. It is not active with 4SB and CELP coding algorithms.

5. Use cases

The table below gives some concrete example cases with actions that can be taken to compensate for quality problems or simplify use.

Problem or situation	Symptoms	Actions	Notes and supplements
Numerous packet losses	Numerous audio dropouts. Network quality indicator: low quality. Detailed statistics: many lost packets.	Reduce the bit rate	Opus allows you to modify the bitrate in real time and monitor the effect on quality.
		Activate packet duplication on the *remote* codec	Can be counterproductive if the problem is network congestion. Combine with bit rate reduction and prefer high efficiency encoding: HE-AAC, Opus
		Make use of Double Streaming in "diversity" mode	Withstands long interruptions on an interface during transmission, even on a total loss. The remote codec must also apply the technique.
Very high jitter on the link	Audio dropouts. Network quality indicator: degraded quality. Detailed statistics: high jitter, many packets discarded or reordered.	Increase the jitter buffer size: more pessimistic "Network quality" setting.	Effective for *reception* only.
The user can't master the optimization adjustments (remote report)		Use the automatic settings: Auto Jitter Buffer and Auto Bit Rate Opus	The resulting adjustments may not be optimal
Insufficient bandwidth on the link	*A priori* information on a bandwidth limitation. Jerky audio. Detailed statistics: very numerous packet losses, too low number of received packets	Reduce the bit rate	Prefer high efficiency coding: HE-AAC, Opus. Opus allows you to modify the bitrate in real time and monitor the effect on quality.
		Make use of Double Streaming in "bonding" mode	The remote codec must also apply the technique. Each of the links must be stable.
A low and/or stable latency is required		Prefer Opus for a lower latency. Don't use the Auto Jitter Buffer system.	