**ABSTRACT**

Voice-over-IP (VoIP) call streams suffer from poor quality when the available bandwidth drops below the minimum requirements for the call. Dynamic VoIP (DVoIP) improves quality by performing on-the-fly transformations on VoIP streams to reduce the bandwidth requirements of the call. DVoIP reduces bandwidth requirements by transcoding high-bitrate codecs to low-bandwidth ones and by aggregating the contents of multiple VoIP packets into a single DVoIP packet. With little to no increase in latency or decrease in voice quality, DVoIP can ensure zero packet loss for active calls.

**INTRODUCTION**

Voice-over-IP (VoIP) enables voice calls to be routed across heterogeneous networks using commercial, off-the-shelf (COTS) equipment. Because this requires interoperability among diverse vendors, VoIP protocols and devices cannot be quickly adapted to perform well in all network scenarios. When changing network conditions cause a decrease in the amount of bandwidth available for VoIP, and the VoIP infrastructure cannot dynamically adjust to the new network parameters, packet losses occur due to congestion.

VoIP packet losses result in decreased call quality and speech intelligibility. For speech encoded with the G.711 codec, a 4.9% packet loss rate can significantly reduce call quality, while a comparable loss in quality can occur for the G.729 codec with only a 0.33% loss rate [3]. It is therefore important, especially for low-bandwidth codecs such as G.729, to minimize packet loss rates. Packet losses can cause severe degradation in speech intelligibility, rendering low-bandwidth links such as those used for residential Internet connections useless for carrying VoIP calls.

To reduce these packet losses, we propose Dynamic VoIP (DVoIP), a system that can perform on-the-fly transformations to VoIP streams to ensure smooth and stable operation over a variety of network conditions.

DVoIP improves VoIP call quality of service by reducing the bandwidth requirements of a given VoIP stream, which allows the call stream to pass across a bandwidth-constrained channel unimpeded. This process also allows more VoIP calls to share a channel. DVoIP effects bandwidth adaptation through two actuators: the audio transcoder and the frame aggregator. Audio transcoding is the translation between audio codecs, which allows the system to convert traffic being sent from the VoIP application using a high-bandwidth codec into a lower bandwidth codec suitable for transmission over a constrained link. Frame aggregation, coupled with a specialized Real-time Transport Protocol (RTP) header compression, allows further bandwidth reduction by compressing sequential audio frames into a single packet, reducing network overhead through a negligible increase in stream latency.

Either one or both of these actuators can be used at any given time, and each has parameters that can be adjusted to achieve the desired bandwidth and minimize unnecessary signal loss and latency increase. The audio transcoder can convert between any two audio codecs in its library, and the frame aggregator can be configured to include an arbitrary number of audio frames in a single packet. These parameters can be modified on-the-fly to adjust to changing network conditions without interrupting in-progress calls.

DVoIP operates at the network layer (ISO OSI Layer 3), meaning that DVoIP can be incorporated into a router, a switch, a network accelerator appliance, a VoIP Private Branch Exchange (PBX), or even the IP networking stack of an operating system. By operating at the network layer, DVoIP is transparent to the application and can be used with any RTP-based VoIP application.

The transformations made by DVoIP are reversible. To maintain application transparency, any transformations performed on the stream at one endpoint must be reversed before passing the recovered packet stream to the application. A prototypical example of DVoIP includes two VoIP Application endpoints, and two DVoIP systems, each of which includes an encoder/decoder pair (Figure 1). The two DVoIP systems sit between the application and the

---

constrained network link. Each DVoIP system encodes the packets sent by its local VoIP application and simultaneously decodes the packets sent from the remote DVoIP system. Because the network conditions may differ in each direction, we do not necessarily want to use the same parameters for each end of the call. Thus, DVoIP allows the transformations to be applied asymmetrically at each endpoint.

**RELATED WORK**

Previous work on payload transcoding has focused on enabling access to streaming media by diverse client devices that may not be compatible with the protocols or codecs used by the server [4, 5, 6]. In these cases, one or more proxies mediate the connection between the client and server, performing transformations on the stream as needed for the client to receive data that it can decode and process. Though there may be a resultant reduction in bandwidth requirements, the goal of these techniques is interoperability rather than bandwidth conservation.

Transcoding has also been proposed as a method to enable access to streaming audio over bandwidth-limited links. Though [1] suggests that audio and video can be transcoded to formats that are smaller and potentially have lower quality, the authors do not propose a system to handle the transcoding process. However, this work does provide a number of results concerning the effect on quality of using various audio and video codecs for transcoding.

Frame aggregation is shown to be an effective means to reduce the bandwidth consumption due to packet overhead, thus increasing the number of calls that can be carried by a link. Garg and Kappes [4] demonstrate that increasing payload sizes (the number of audio frames) can dramatically increase the number of calls that can be carried by a channel, while [7] describes a technique for aggregating VoIP packet payloads to reduce overhead. The process that Yun et al. [7] discuss operates at Layer 2 and is designed to reduce IEEE 802.11b network overhead ignoring the Internet Protocol/User Datagram Protocol/Real-time Transport Protocol (IP/UDP/RTP) overhead as insignificant in comparison. In contrast, DVoIP operates at Layer 3 and would be useful even when applied along with Yun et al.’s work. Further, [7] is not transparent to the VoIP application and could require modification to the application to be effective. None of the papers mentioned above address the use of transcoding for providing similar quality of service at a reduced bandwidth. Further, the application of frame aggregation to VoIP calls is only mentioned in the context of 802.11 wireless networks, which are relatively high-bandwidth links. By combining the features of media transcoding and frame aggregation, DVoIP yields significant quality of service improvement for VoIP calls in constrained, low-bandwidth scenarios.

**DVoIP DESIGN**

Several steps are involved in DVoIP processing. First packets are received from the VoIP application by the DVoIP encoder. The encoder carries out its audio transcoding and frame aggregation step and then passes the transformed audio frames to the DVoIP packet encoder stage. The packet encoder builds a DVoIP packet containing the transformed audio, all the original RTP headers, and a special DVoIP header that directs the decoding of the packet and reassembly of the original VoIP packets at the receiving side. This header indicates how many audio frames are included in the packet, and for each frame, whether it came from a single original packet or from more than one packet, which codec is used to encode the frame and the position in the original packet of the frame.

At the receiving side, the packet stream is passed to the DVoIP decoder. Each packet is decoded to determine how many audio frames are included, which codec was used to encode them, and to which original packet they belong. The audio transcoder is used to convert the frames back to the original audio codec, the re-encoded frames are reassembled into packets, and the original RTP headers are reconstructed and attached to the packets. Finally, the rebuilt stream is sent to the receiving VoIP application.
A. Audio Transcoder

The DVoIP audio transcoder converts audio from one codec to another, allowing the system to make a tradeoff between speech quality and bandwidth requirements. If the VoIP application transmits packets containing a high-bitrate, high-quality codec, the transcoder can change the codec to a lower-bitrate, lower-quality codec to save bandwidth. The transcoder includes a library of codec plugins that translate between that codec and standard 16-bit Pulse Code Modulation (PCM) audio format. The codec library is necessary for two reasons. First, it is important to be able to decode audio in the format used by a VoIP application, and second, we need a collection of low-bitrate codecs to be able to transmit audio over the constrained link.

To transcode a VoIP packet, first we decode all the audio in the packet into 10 ms frames of 16-bit PCM audio. At the standard narrowband sampling rate of 8000 Hertz (Hz), 10 millisecond (ms) corresponds to 80 samples. The number of PCM frames obtained depends on the frame size (in samples) of the original codec and the number of frames encoded in the packet. Each codec includes a lookup table that describes the required frame size in samples and bytes, the preferred number of frames to include in a packet, and other useful parameters.

Once we have the audio in PCM format, we can reencode these frames with our target codec. Because the samples-per-frame parameter of our original codec may not be the same as our target codec, we determine the number of target codec frames we can encode with the PCM frames we have available. Due to the differences between samples-per-frame and frames-per-packet between the two codecs, we may have leftover PCM frames that could not be encoded in this packet. These PCM frames will be retained in a buffer, along with information about the packet that contained these audio frames, including the original RTP header. If packets in the buffer reach a timeout threshold, they will be flushed and transmitted to the remote side so they can be sent to the application before they expire. Each PCM frame encoded in a DVoIP packet will carry its original packet information with it as part of the DVoIP packet metadata.

B. Frame Aggregation

For further reductions in bandwidth requirements, DVoIP uses frame aggregation to combine the audio frames from multiple packets into one packet. Frame aggregation trades audio frame latency for a reduction in header overhead and, thus, a reduction in bandwidth. A typical VoIP scenario involves an application that transmits 50 packets per second, each one containing 20ms of audio data. Each packet includes an IP/UDP/RTP header combination totaling 40 bytes. This yields an overhead of 16000 bits per second, which, depending on the codec, can be larger than the audio data itself (Figure 2).

![Codec Bandwidth Requirements and Packet Overhead (50 packets/s)](image)

By combining multiple VoIP packet payloads into a single packet, we can eliminate the IP/UDP overhead for all but one of the packets. We cannot simply discard the RTP headers associated with each packet, but we can compress them from 12 bytes to an average of 2.5 bytes each.

This module accepts a frame count parameter that controls the number of frames to be aggregated into a single packet. When frame aggregation is enabled, audio frames are buffered until there enough frames to equal the frame count. Frames are combined into a single packet, and each frame is tagged with appropriate header information to direct the unpacking and reassembly process at the receiving end. Because each audio frame has RTP header information associated with it, we use our RTP header compression to reduce the burden of our header information on packet overhead.

RESULTS

A. Analytical Results

To evaluate performance, analytical results for DVoIP were computed assuming a 128 kilobits per second (kbps) link. The values presented include overhead due to IP, UDP, and RTP headers, as well as additional overhead incurred by DVoIP headers. Prototype DVoIP system overhead is approximately 3.5 bytes per 10 ms audio frame that is included in the packet. These results do not include overhead due to link layer protocols. As the per-packet overhead increases, DVoIP can make even larger gains in bandwidth reduction than are presented here.
For an individual VoIP call, DVoIP can increase quality of service by reducing the bandwidth requirements of the call. Assuming a source VoIP stream that produces 20 ms packets of G.729-encoded speech, we will encounter 50 packets per second. This yields a total bandwidth requirement of 24 kbps, including IP overhead.

Figure 3 shows the decrease in bandwidth for a G.729 call due to increasing amounts of frame aggregation, as well as transcoding to lower bitrate audio codecs. In the case of Speex 8k and Speex 6k, when no frame aggregation is used, the overhead due to DVoIP causes the bandwidth to exceed the total bandwidth of the original G.729 stream. In all other cases, the bandwidth of the calls have been considerably reduced in even with just a small amount of frame aggregation. Further gains are made by transcoding the audio packets to a low bandwidth codec. For example, if we transcode the call to Speex 4k and increase frame aggregation to four frames (i.e., 80 ms of audio), we reduce the bandwidth to less than half of the original, unmodified G.729 stream.

DVoIP also allows us to increase the number of calls that can be carried by a channel. Ignoring Link Layer overhead, if we assume 128 kbps of bandwidth is available, we can fit five concurrent G.729 calls into the channel. As in the example above, if we apply a frame aggregation rate of four frames and transcode to Speex 4k, we can more than double the number of calls that can be carried by the channel (Figure 4).

**B. Experimental Results**

We have produced a prototype implementation of DVoIP that can interface with both hardware and software VoIP terminals. To test DVoIP performance, we have designed a testbench application that replays VoIP streams that have been recorded from Cisco 7940 IP phones. The testbench generates multiple concurrent VoIP streams. These streams are passed through a Linux-based router with a link constraint of 128 kbps. At either end of the link we have a DVoIP endpoint capable of handling multiple simultaneous channels.

We have generated results from the experimental setup above, in which 10 G.729 calls are placed across a link that has been constrained to 128 kbps. Each G.729 call requires 24 kbps, yielding a requirement of 240 kbps for all 10 calls. Given a link constraint of 128 kbps packet loss due to congestion and jitter because of queue processing time will occur. Packet loss will reduce speech intelligibility, and increased jitter will cause further impairment. Figure 5 shows a large variation in bandwidth usage across all 10 calls, indicating congestion. Furthermore, none of the calls reach a full 24 kbps of bandwidth, meaning that there is constantly some packet loss happening for all calls.

If we place a DVoIP system at either side of the constrained link, we can transform the original G.729 streams to fit easily within the allotted bandwidth. By transcoding each call from G.729 to Speex 4k, and aggregating every 80ms of speech into a single packet, we can reduce per-call bandwidth to 10.7 kbps. Figure 6 shows the per-call bandwidth of all 10 calls when the above DVoIP transformations are applied to the stream. The lack of variation indicates that there is no bandwidth contention, and thus we have alleviated congestion loss and jitter.
A single G.729 call requires 24kbps of bandwidth to operate without any loss, meaning that 240kbps would be necessary to carry all ten calls. In this scenario, none of the calls receives 24 kbps of bandwidth, which means each call stream is subject to constant loss. The large variation in bandwidth for each call indicates that channel congestion is causing a large amount of loss and jitter for each call.

Figure 5. Per-call bandwidth of ten simultaneous G.729 VoIP calls being carried over a 128 kbps link.

Each G.729 stream transcoded to the Speex4 kbps codec, while being aggregated into 80 ms frames of audio data, yielding a bandwidth requirement of 10.7kbps per call. The amount of bandwidth required for all ten calls is well within the limits of the 128 kbps channel, as indicated by the smoothness of the per-call bandwidth graph.

Figure 6. Per-call bandwidth of ten simultaneous G.729 calls with DVoIP being carried over a 128 kbps link.

We have presented the theory and design of DVoIP, detailing the components involved and their methods of operation. DVoIP reduces bandwidth requirements for VoIP calls, allowing better performance for individual calls in network scenarios where bandwidth is constrained, as well as fitting more calls into a given amount of bandwidth. We have demonstrated a twofold increase in the call volume that can be handled by a 128 kbps link when using DVoIP.

**FUTURE WORK**

When applying frame aggregation to an audio stream, we are transmitting fewer packets that contain more audio frames. If the network link between DVoIP nodes is subject to errors and packets are dropped, we will lose more data if frame aggregation is enabled than if we used only audio transcoding. This could cause serious disruptions to the VoIP call. To avoid this situation, we will investigate adding forward error correction (FEC) to DVoIP packets in order to mitigate the effect of packet loss on the VoIP call. Our future work will explore the differences between techniques and the tradeoff of amount of redundancy versus the added bandwidth overhead due to forward error correction.

**REFERENCES**


