End-to-End Delay Estimation without Synchronized Clocks

Bernd Deffner and Henning Schulzrinne
GMD Fokus
10623 Berlin, Germany
schulzrinne@fokus.gmd.de

June 8, 1996

Abstract

Interactive audio applications such as telephony or teleconferencing are highly sensitive to end-to-end delays. These delays consist of the time for compression, network propagation delays, the playout delay and the playback delay. These delay components cannot be easily estimated, particularly if audio or video sender and receiver do not share a closely synchronized clock. This note presents some experiments on how to accurately measure delays from sound generation to production, either locally or across wide-area networks such as the Internet.

1 Introduction

Interactive distributed audio applications such as Internet telephony, audio/video conferencing or interactive games are highly sensitive to end-to-end delays. For example, smooth conversation becomes difficult if the end-to-end delay exceeds about 200 ms [1], while echo acoustic effects become annoying at delays above about 40 ms.

The end-to-end delay consists of a number of components. We assume here that the audio interface in a workstation digitizes a block of audio samples before sending an interrupt to the operating system. The OS then copies these samples to a buffer and signals to the application that audio data is available for reading.

Acquisition delay: The acquistion delay is the time from when the sound reaches the microphone to the time the user software receives the block of samples. This delay is at least as long as the audio block length, with some additional overhead due to copying and context switches.

Compression delay: The compression delay is the delay introduced by any software compression performed by the sender.

Propagation delay: The propagation delay is the speed-of-light delay between sender and receiver, typically about 5 μs per km for fiber connections. (There may be other fixed delay due to the time it takes to transmit packets, but an observer cannot distinguish these from the propagation delay.)

Decoding delay: Each arriving packet has to be converted back to an uncompressed audio format before it can be played out, unless this is done by the audio hardware.

Queueing and playout delay: Due to variable queueing delays in the network, the delay of individual packets will vary (delay jitter). In order to play out audio packets with the same periodic spacing as they had at the source, the receiver uses a playout buffer that inserts additional delay so that the sum of the queueing and playout delay is a constant delay[1], roughly equal to the maximum delay jitter [2].

Playback delay: Audio blocks emerging from the playout buffer are written to the audio output device and, after some additional buffering provided by the operating system, reach the D/A converter.

With timestamps and synchronized clocks, all but the first and last of the delays just enumerated can be estimated. However, the acquisition delay and playback delay are not easily measured. Also, while clock synchronization is widely available [3], synchronizing clocks to the required accuracy of a few milliseconds is not readily available across the Internet. Even if both sender and receiver workstation use a GPS or other time-signal receiver, getting timing information with millisecond accuracy is not easy, particularly with typical non-real-time operating systems [4]. Measurements based on timestamps also require access to the source code of the audio software, something that is usually not available for commercial products.

Typically, network delays are measured as round-trip delays, e.g., by programs like `ping` and `traceroute` or the RTCP [5] sender and receiver reports. Measuring round-trip delays does not require synchronized clocks. However, network delays can be highly assymetric, due to different routes [6] or differing network congestion.

This note presents an alternative measurement approach that allows to accurately estimate end-to-end delays, without requiring synchronized clocks, symmetric paths or source code access. Our approach is based on the phase difference of audio signals.

2 Measuring Audio Delays through Correlation

Audio delays can be measured with two synchronized audio sources and a workstation capable of recording stereo audio signals. We call audio sources 'synchronized' if they both produce the same signal with a known phase delay between them. The workstation records a stereo audio signal, with one channel recorded from the local audio source and the other channel from the remote audio source being transmitted over the network. The phase difference between the two channels reflects the overall delay from sender to receiver and can be computed with the cross correlation function. Figure 1 shows the measurement scenario with two radios as audio sources.

We used NEVOT as the audio sending application at host 1 and as the receiving application at host 2. At host 3 a small program is used to record the resulting stereo audio signal and store it in a file. This file can then be analyzed by applying standard correlation techniques, either in the time or frequency (FFT) domain. As can be seen from comparing Fig. 3, where female speech was used, to Fig. 7, with a white noise source, the periodic components of speech and music introduce a number of spurious peaks into the correlation function.

3 Future Work

We have currently used only radios receiving from the same transmission station, with the receivers located close to each other. For wide-area measurements, including those spanning continents, the challenge is to find two audio sources that have a known timing relationship. Unfortunately, the audible WWV time signals at 5, 10 and 15 MHz shortwave cannot be readily received in Europe. A number of other possibilities have been explored.

Most shortwave stations broadcast a five-tone time signal and the common hourly news, even though the programs themselves tend to be regionalized. For accurate measurements, one needs to know the propagation

[1] at least for the duration of a talkspurt

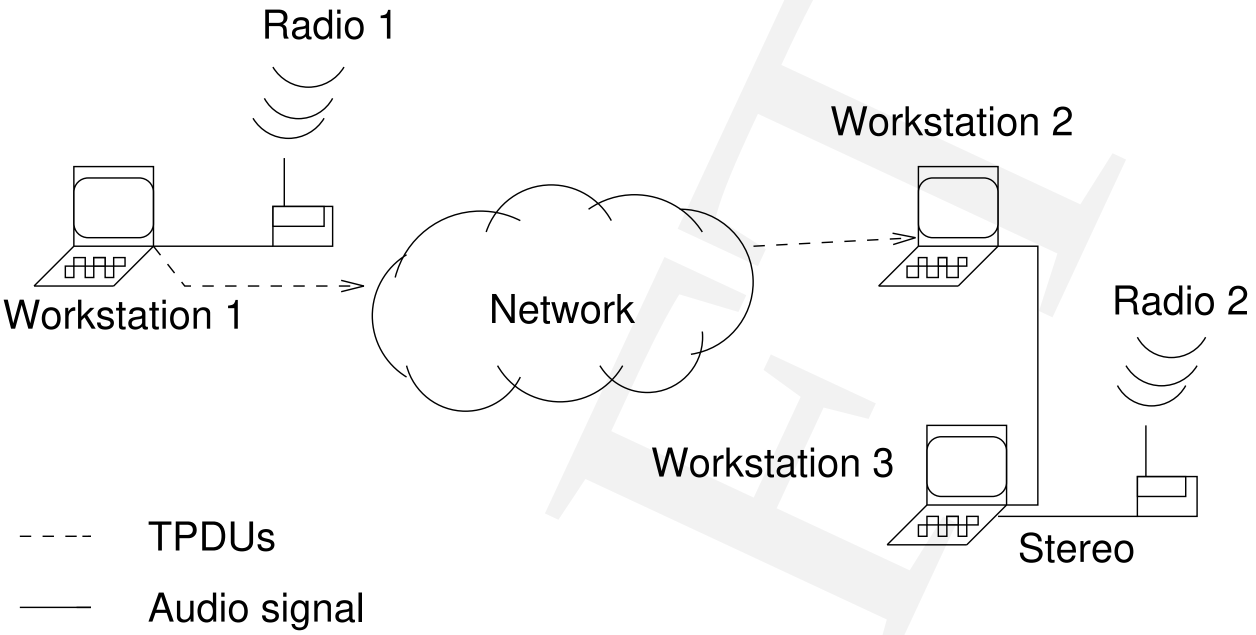

Figure 1: Measuring delays with two synchronized audio sources

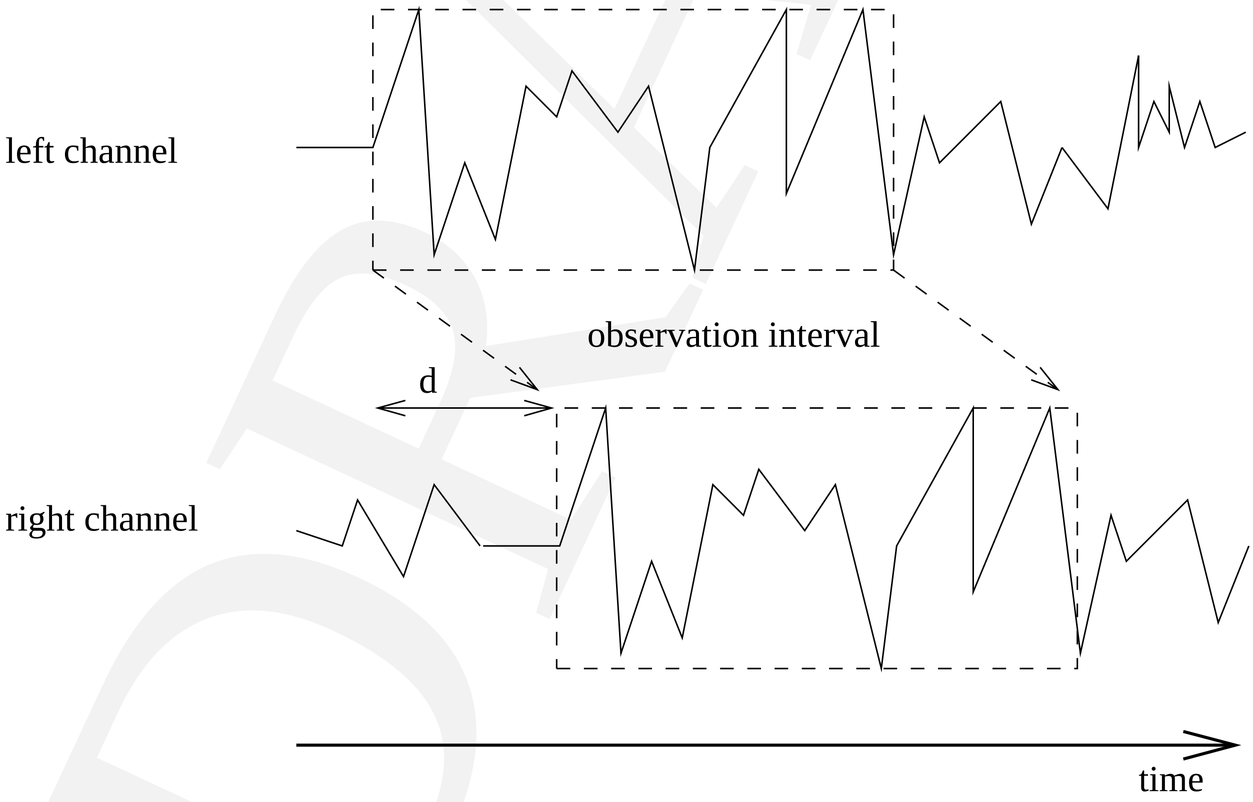

Figure 2: The stereo audio signal

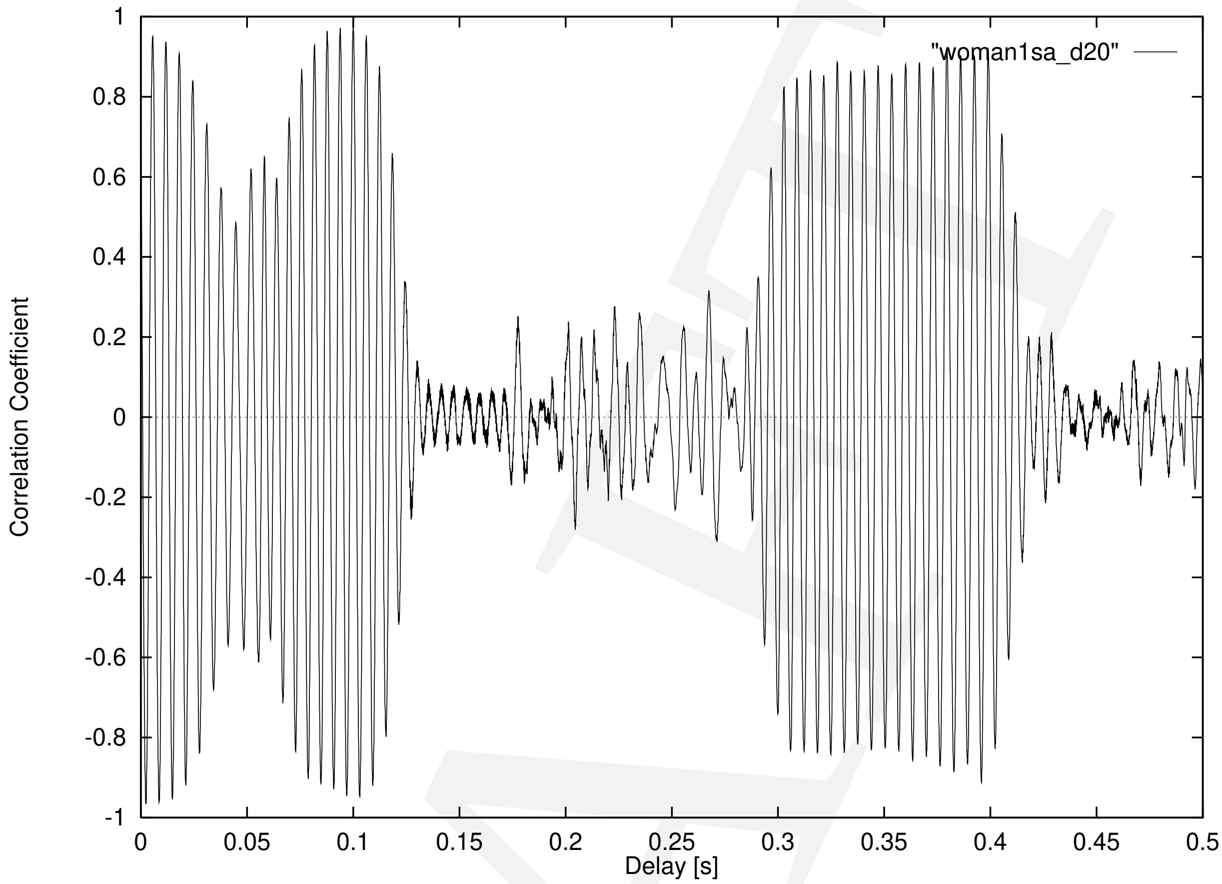

Figure 3: Correlation: one radio, speech, artificial delay of 100 ms

delay, which may also depend on where the signal is reflected in the stratosphere or whether the signal is propagated by ground wave.

References

[1] T. M. Chen, J. Walrand, and D. G. Messerschmitt, "Dynamic priority protocols for packet voice," *IEEE Journal on Selected Areas in Communications*, vol. SAC-7, pp. 632–643, June 1989.

[2] R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne, "Adaptive playout mechanisms for packetized audio applications in wide-area networks," in *Proceedings of the Conference on Computer Communications (IEEE Infocom)*, (Toronto, Canada), pp. 680–688, IEEE Computer Society Press, Los Alamitos, California, June 1994.

[3] D. L. Mills, "Internet time synchronization: the network time protocol," *IEEE Transactions on Communications*, vol. 39, pp. 1482–1493, Oct. 1991.

[4] D. L. Mills, "Precision synchronization of computer network clocks," *ACM Computer Communication Review*, vol. 24, pp. 28–43, Apr. 1994.

[5] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: a transport protocol for real-time applications," Request for Comments (Proposed Standard) RFC 1889, Internet Engineering Task Force, Jan. 1996.

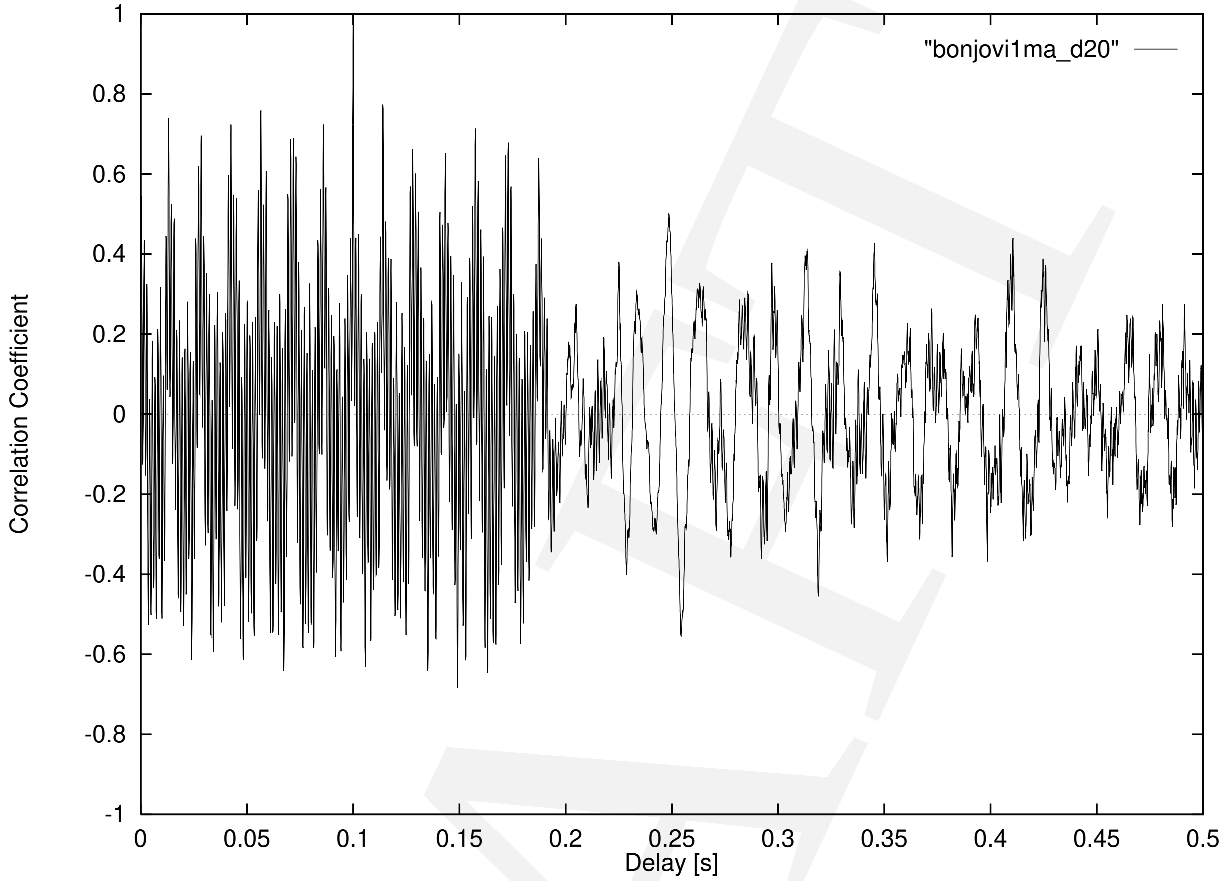

Figure 4: Correlation: one radio, music, artificial delay of 100 ms

[6] V. Paxson, "End-to-end routing behavior in the internet," in *SIGCOMM Symposium on Communications Architectures and Protocols*, (Stanford, California), Aug. 1996.

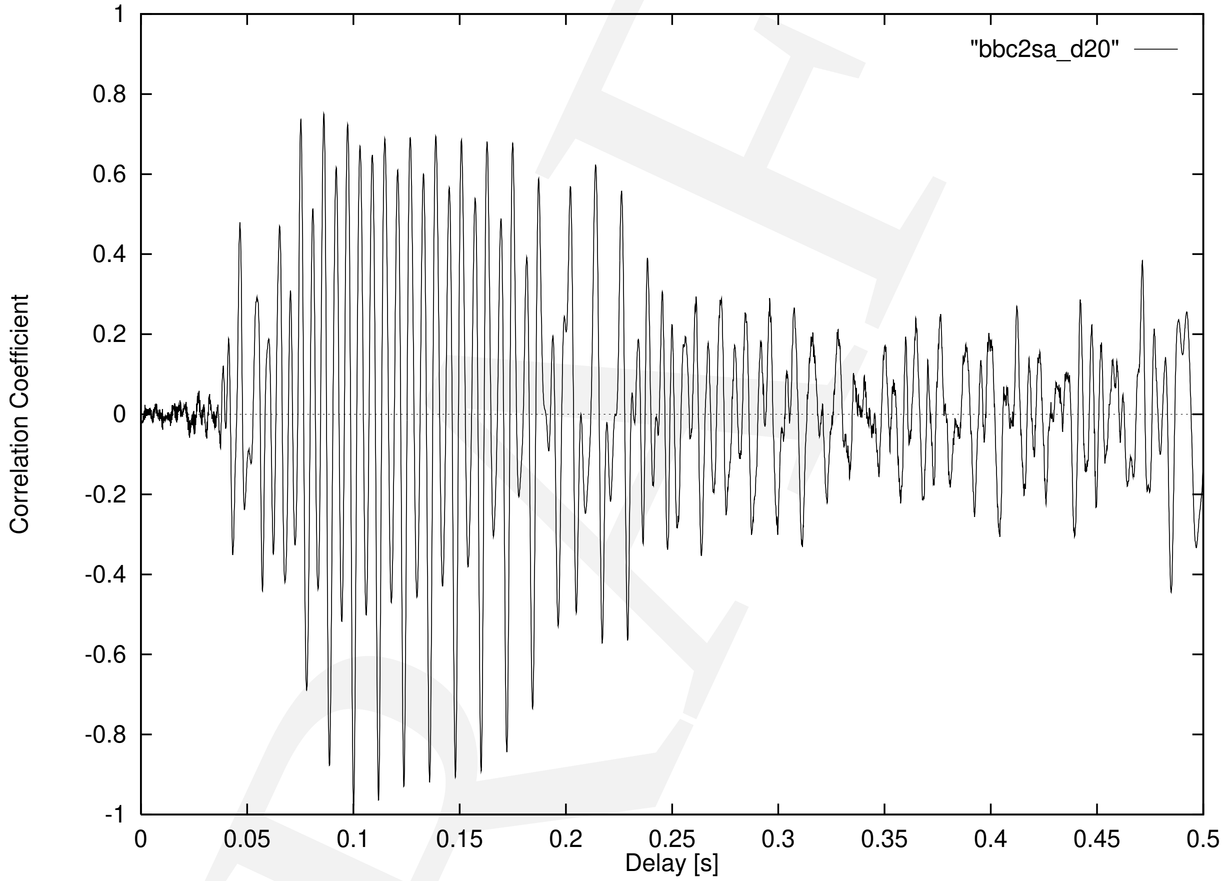

Figure 5: Crosscorrelation: two radios, speech, artificial delay

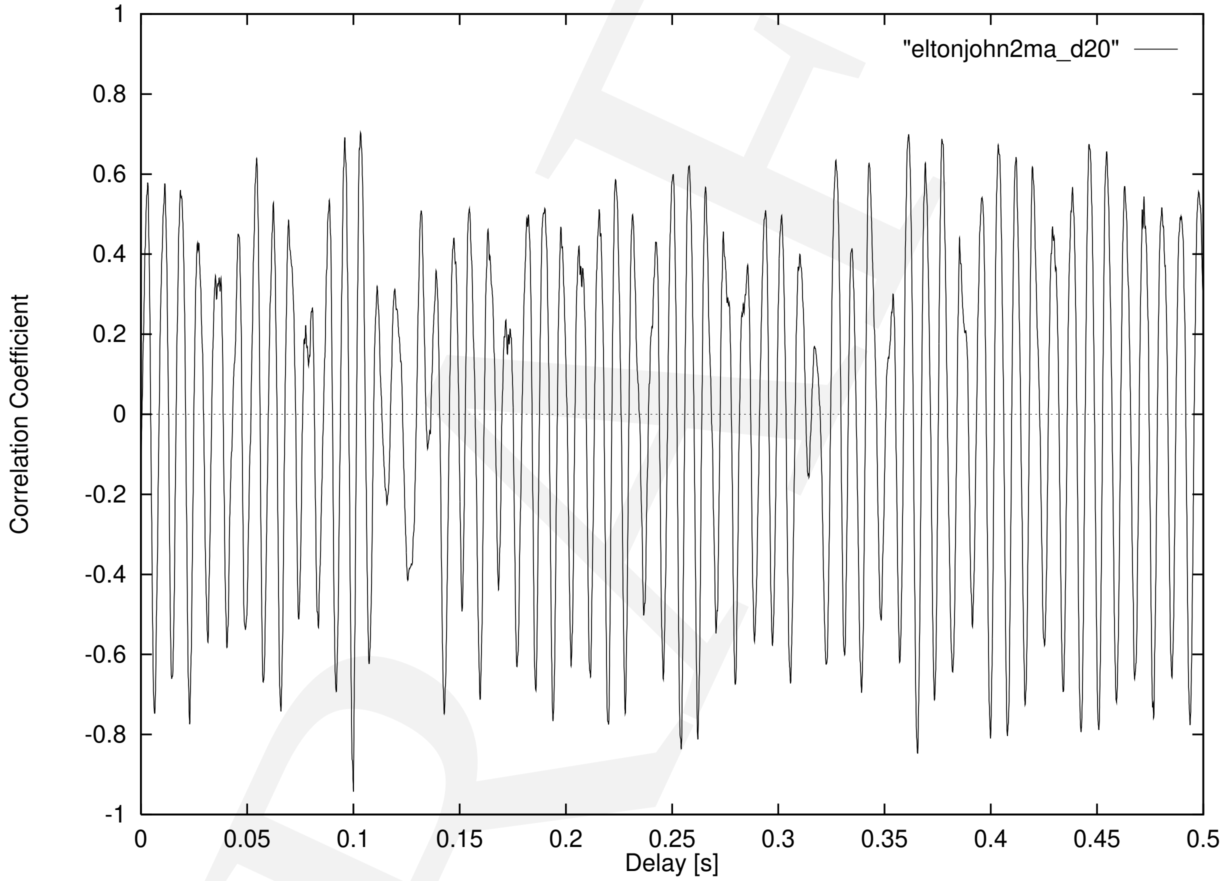

Figure 6: Crosscorrelation: two radios, music, artificial delay of 100 ms

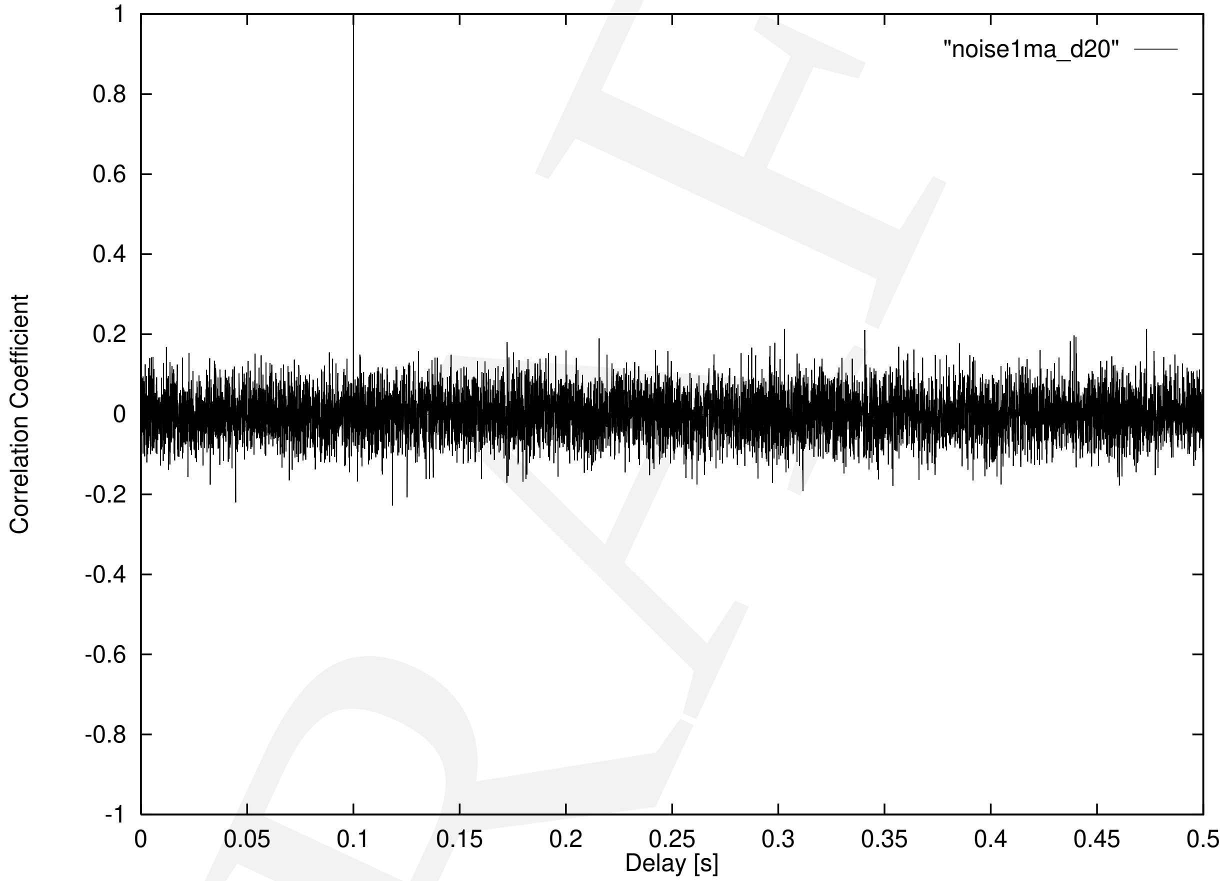

Figure 7: Crosscorrelation: one noise source, artificial delay of 100 ms

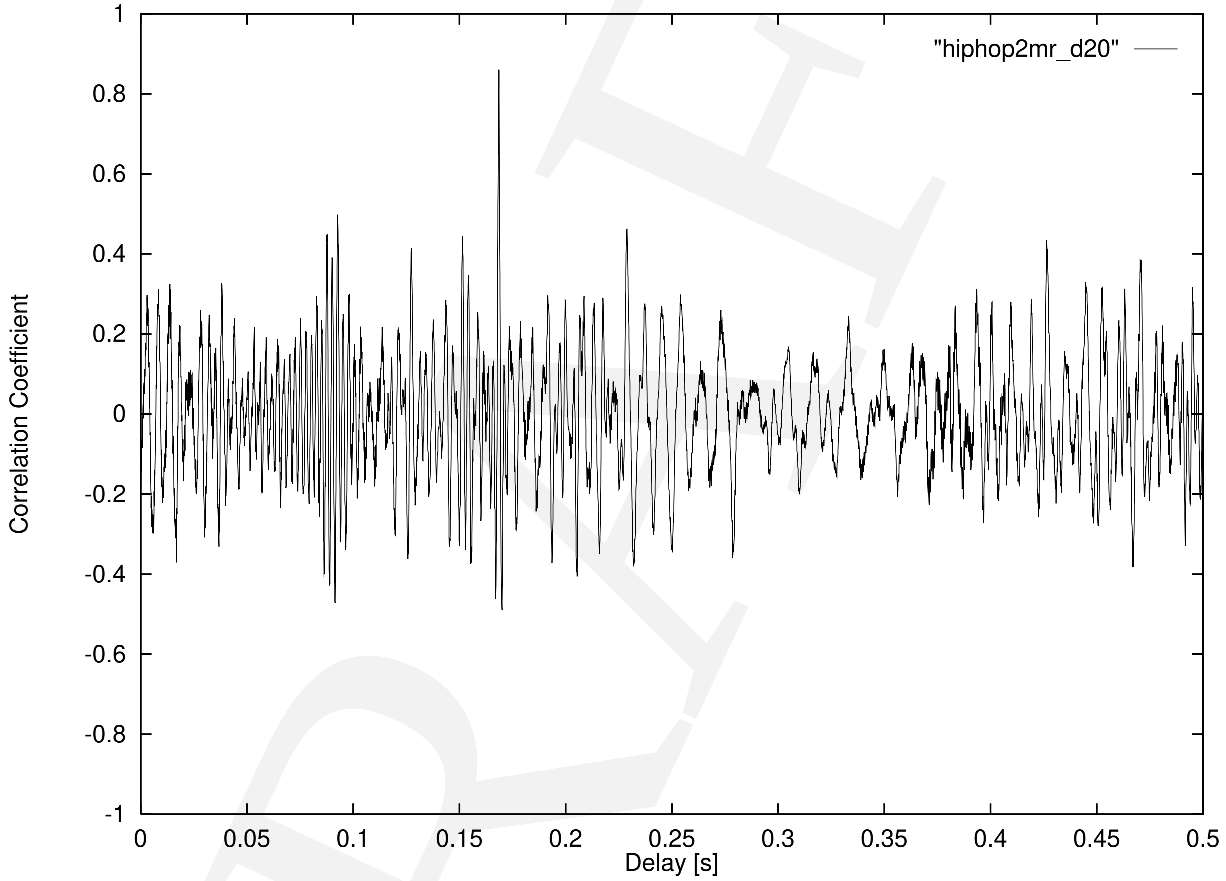

Figure 8: Crosscorrelation: two radios, music, real delay

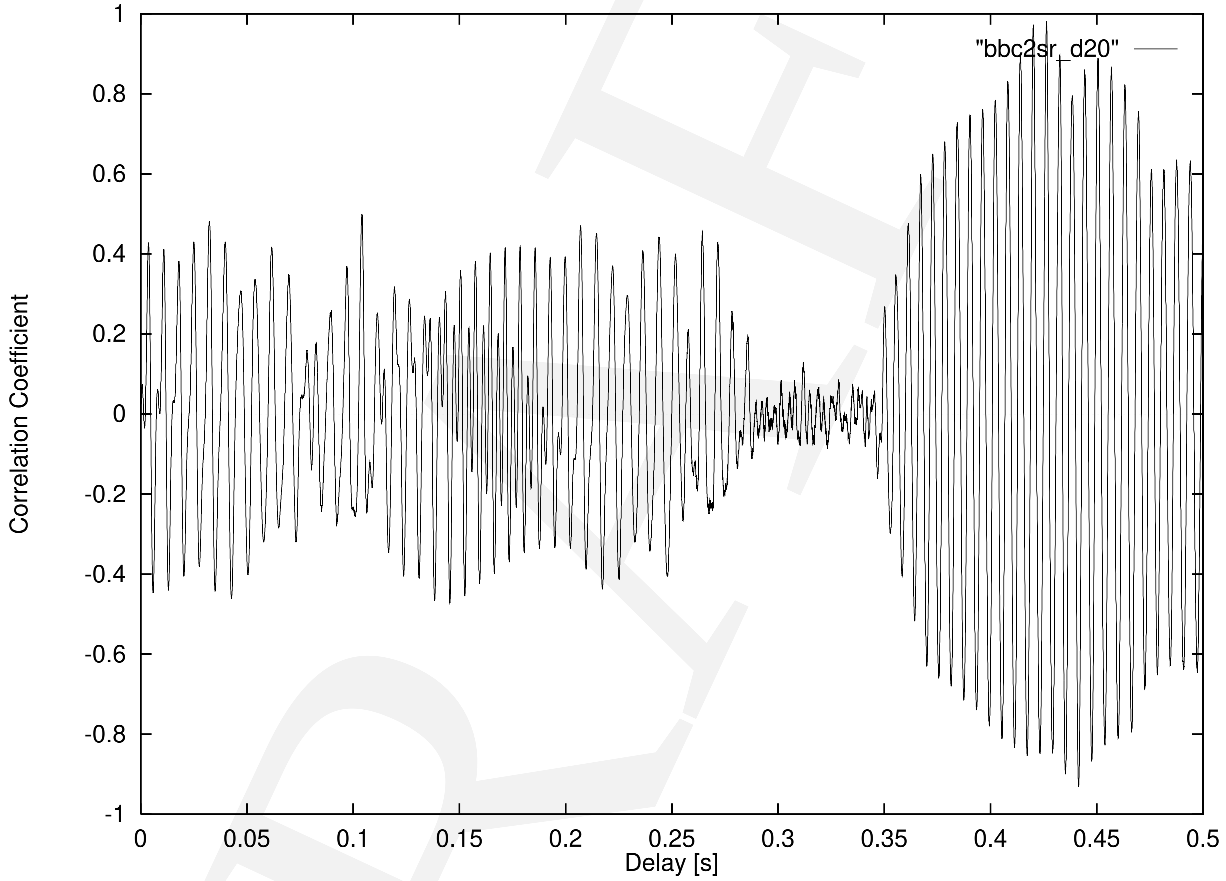

Figure 9: Crosscorrelation: two radios, real delay