

Time and Phase Delay in Loudspeakers Playback

Yangang Cao

March 5, 2019

When a constant delay is applied to one loudspeaker in stereophonic listening, virtual sources with transient signals are perceived to migrate towards the loudspeaker that radiates the earlier sound signal. Maximal effect is achieved asymptotically when the delay is approximately 1.0 ms or more. However, the effect depends on the signal used. With continuous signals containing low frequencies, the effect is much less prominent than with modulated signals containing high frequencies.

In such processing the phase or time delays between the loudspeakers are turned at low frequencies into level differences between the ears, and at high frequencies to time differences between the ears. This all makes the virtual source direction depend on frequency. The produced binaural cues vary with frequency, and different cues suggest different directions for virtual sources. It may thus generate a “spread” perception of direction of sound, which is desirable in some cases. The effect is dependent on listening position. For example, if the sound signal is delayed by 1 ms in one loudspeaker, the listener can compensate the delay by moving 30 cm towards the delayed loudspeaker.

```
1 % Creating spatially spread virtual source by delaying one channel
2
3 Fs=44100;
4 % Delay parameter for channel 1 in seconds
5 delay=0.005;
6 % Corresponding number of delayed samples
7 delaysamp=round(delay*Fs)
8 % Signal to be used
9 signal=mod([1:20000]',400)/400;
10 signal(1:2000)=signal(1:2000).*[1:2000]'/2000; % Fade in
11 % Delaying first channel
12 loudsp_sig=[[zeros(delaysamp,1); signal(1:end-delaysamp)] signal];
13 % Play audio with loudspeakers
14 soundsc(loudsp_sig,Fs);
```

A special case of a phase difference in stereophonic reproduction is the use of antiphasic signals in the loudspeakers. In such a technique, the same signal is applied to both loudspeakers, however, the polarity of the other loudspeaker signal is inverted, which produces a constant 180° phase difference between the signals at all frequencies. This changes the perceived sound color, and also spreads the virtual sources. Depending on the listening position, the low frequencies may be cancelled out. At higher frequencies this effect is milder. This effect is also milder in rooms with longer reverberation. The directional perception of the antiphasic virtual source depends on the listening position. In

the sweet spot, the high frequencies are perceived to be at the center, and low frequencies in random directions. Outside the sweet spot, the direction is either random, or towards the closest loudspeaker. In the language of professional audio engineers this effect is called “phasy”, or “there is phase error in here”.

```

1 % Create a spread virtual source by inverting phase in one ...
  loudspeaker
2
3 Fs=44100;
4 signal=mod([1:20000]',400)/400; %signal to be used
5 % Inverting one loudspeaker signal
6 loudsp_sig=[-signal signal];
7 % Play audio out with two loudspeakers
8 soundsc(loudsp_sig,Fs);

```

A further method to spread the virtual source between the loudspeakers is to change the phase spectrum of the sound differently at different frequencies. A basic method is to convolve the signal for the loudspeakers with two different short bursts of white noise. Another method is to apply a different delay to different frequencies. This effectively spreads out the virtual source between the loudspeakers, and the effect is audible over a large listening area. Unfortunately, the processing changes the temporal response slightly, which may be audible as temporal smearing of transients of the signal.

Below is a example creating spread virtual sources for stereophonic listening by convolving the sound with short noise bursts:

```

1 % Example how to spread a virtual source over N loudspeakers
2
3 Fs=44100;
4 signal=mod([1:20000]',400)/400; % Signal to be used
5 NChan=2; % Number of channels
6 % Generate noise bursts for all channels
7 nois=rand(round(0.05*Fs),NChan)-0.5;
8 % Convolve signal with bursts
9 loudsp_sig=conv(signal,nois(:,1));
10 for i=2:NChan
11 loudsp_sig=[loudsp_sig conv(signal,nois(:,i))];
12 end
13 if NChan == 2
14 % Play audio out with loudspeakers
15 soundsc(loudsp_sig,Fs);
16 else
17 % Write file to disk
18 loudsp_sig=loudsp_sig/max(max(loudsp_sig))*0.9;
19 wavwrite([loudsp_sig],Fs,16,'burstex.wav');
20 end

```