Homework 3:
LESLIE SPEAKER EMULATION

Sound Analysis Synthesis and Processing
Module II: Sound Synthesis and Spatial Processing

Albarracin Juan Camilo
10817671
Curcio Lorenzo
10844919

POLITECNICO DI MILANO
2023

1 Leslie speaker effect implementation

For this homework, it was required to implement the digital scheme in figure 1.

1.1 Implementation scheme

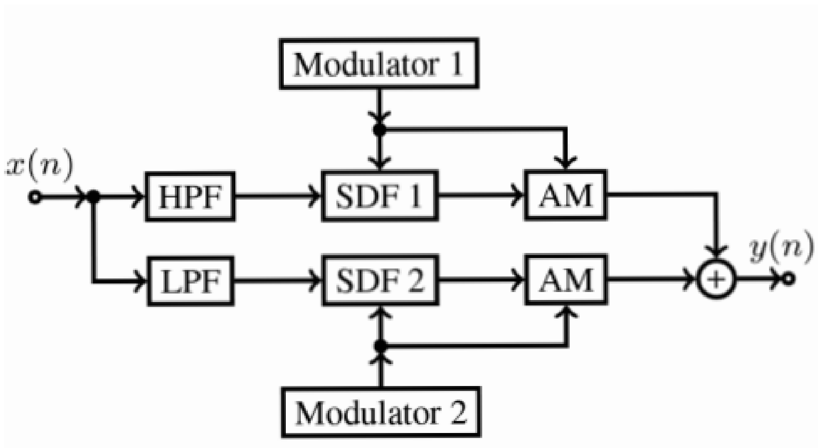

Figure 1: Block diagram for Leslie speaker emulation

Here the crossover is emulated using two 4th order Butterworth filters, in particular they are a high pass filter and a low pass one, with cutoff frequencies equal to 800 Hz. The spectral delay filters simulate the frequency modulation of both the bass and treble bands of the signal. due to the Doppler effect generated by the rotating components (the drum and the horn). Due to the same motions, an amplitude modulation is also to be introduced, using modulators with the same frequencies. Finally, the two components are recombined to produce the desired signal.

1.2 Implementation of the blocks

To make it possible for this scheme to be used in real time, one sample per cycle is computed, this means the Butterworth filters and the SDFs are implemented as difference equations. For the crossover, considering the 4th order filters, whose coefficients were computed using the native MATLAB function:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{4} a_k z^{-k}}{\sum_{k=0}^{4} b_k z^{-k}}$$

each sample was computed as:

$$y(n) = \frac{1}{b_0}\sum_{k=0}^{4} a_k x(n-k) - \frac{1}{b_0}\sum_{k=1}^{4} b_k y(n-k)$$

In order to have access to the values of the samples in the needed previous iterations, buffers were used for both the input and output of each of the 4 filters.

To implement the SDFs, which are of length N = 3,4 respectively for the bass and treble components, the following suggested difference equation was used:

$$y(n) = \sum_{i=0}^{N} \binom{N}{i} m^i(n)[x(n-(N-i)) - y(n-i)]$$

where m(n) is the modulator signal:

$$m(n) = M_s m_0(n) + M_b$$

and $m_0(n)$ is a unitary-amplitude zero-phased sinusoid with frequency 2-6 Hz (depending on which of the two effects is being used) for the bass, and 0.1 Hz higher for the treble (in each effect). M_s and M_b are respectively equal to 0.2 and -0.75 for treble, and to 0.04 and -0.92 for bass.

The amplitude modulation is easily performed as a multiplication, with the following input-output relation:

$$y(n) = [1 + \alpha m(n)]x(n)$$

where $\alpha = 0.9$, buffers are not needed for this step, as the output obviously does not depend on previous samples.

1.3 Results

The resulting signals for both the *Chorale* and *Tremolo* effects were compared with the ground truth ones, and the following MSE error values were obtained:

$$MSE_{ch} \simeq 3.11 \times 10^{-10}$$

$$MSE_{tr} \simeq 3.11 \times 10^{-10}$$

2 Questions

1.

To implement the characteristic vibrato effect of the Leslie amplifier, another feasible approach is to process the instrument's signal using a time-varying delay line controlled by means of a low frequency oscillator (LFO). Considering the industry standard audio effect structure in figure 2, the effect is achieved by setting the *feedback* and *blend* parameters to zero, thus leaving just the *feedforward* path with unitary value.

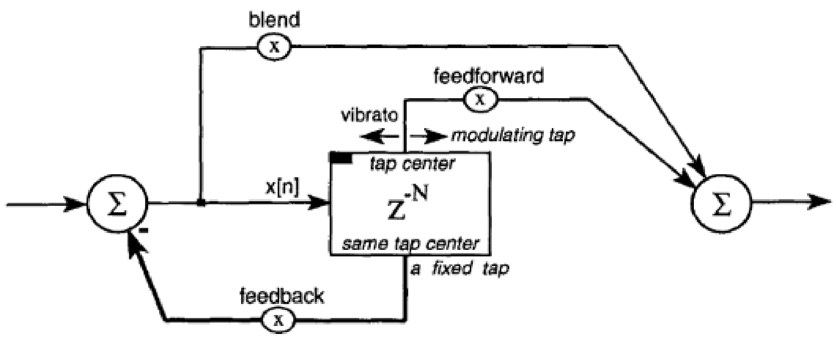

Figure 2: Industry standard audio effect structure

The effect that is obtained depends on the LFO's amplitude and frequency (typically from 0.1 to 5 Hz).

2.

Considering the transfer function of an Nth order SDF, given by a series of allpass filters:

$$H(z) = (\frac{a_1 + z^{-1}}{1 + a_1 z^{-1}})^N$$

The stability condition is:

$$|a_1| < 1$$

In our case, the coefficient a_1 for each SDF is modulated in time, and is equal to $m(n)$. This means the values of M_s and M_b have to be chosen such that the resulting sinusoidal modulator is always in the range [1,-1]. The condition is satisfied with the given parameters since the treble modulator oscillates between -0.95 and -0.55, while the bass one oscillates between -0.96 and -0.88.

3.

The expression for the group delay of a first order SDF is:

$$\tau_g(\omega) = -\frac{\partial \angle H(\omega)}{\partial \omega} = \frac{1 - a_1^2}{1 + 2a_1 cos(\omega) + a_1^2}$$

it is obviously not linear (for $a_1 \neq 0$). Given a signal with two spectral components:

$$u(n) = A_1 sin(\frac{\omega_1 n}{F_s}) + A_2 sin(\frac{\omega_2 n}{F_s})$$

the only instance for which the delay for the two frequencies is exactly equal is obtained when:

$$cos(\omega_1) = cos(\omega_2)$$

However, this is impossible as ω is bounded between 0 and π. A similar group delay can be achieved if the two frequencies both lie in one of the flat regions of the cosine in the said range (near 0 and near π). A pseudo-flat behaviour of the group delay is also achieved for very low values of a_1. However, this limit behaviour makes the group delay necessarily tend to 1 sample as:

$$H_1(z) \simeq z^{-1}$$

$$\tau_{g1}(\omega) \simeq 1$$

4.

For a fractional delay implemented with the Lagrange method, we know the condition to be satisfied is:

$$N > \lfloor D \rfloor$$

Increasing N does not necessarily provide better results in terms of magnitude response and phase delay, in particular the optimal even and odd orders satisfy the following relations respectively:

$$round(D) = \frac{N}{2}$$

$$\lfloor D \rfloor = \frac{N-1}{2}$$

The following MATLAB plots shows the magnitudes and phase delays for different orders. The target plots are a flat magnitude always equal to 1 and a flat phase delay always equal to D = 3.3.

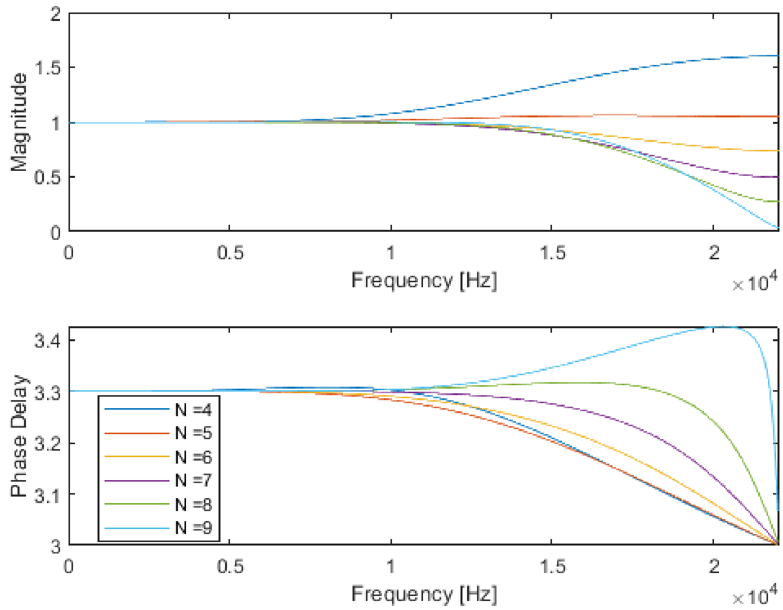

Figure 3: Lagrange method - Magnitude and phase delay responses

As expected, the optimal orders (6 and 7), have overall good performances in both the plots. The 6th order one has better magnitude response, while the 7th order one has better performances in terms of phase response. This is also an expected behaviour of odd and even orders when using the Lagrange method.