Acoustic targets with Duelund functions for multiway loudspeaker

Jean-Marc Plantefève
https://2adiy.ovh

After reiterating the usefulness of defining a set of viable acoustic targets for developing a multiway loudspeaker, Duelund transfer functions are highlighted and a computer-based method for generating targets in "frequency response data" format is proposed.

Crossover networks

When designing crossover networks, one goal is that the combined acoustical output should have a constant amplitude response as a function of frequency. This is achieved when the system is either a constant-voltage type or an all-pass type. A constant-voltage system has a transfer function of unity, that is, its amplitude response is 1 (or any constant) and its phase response is 0 for all frequencies. An all-pass system has a constant amplitude response also, but the phase response is arbitrary. [1]

Erik Baekgaard has shown how adding a band-pass filler driver to a two-way all-pass network turns it into a three-way constant-voltage network. This constant-voltage network may seem ideal as it theoretically transfers any input signal without distortion. However, as the drivers are not in phase for the constant-voltage networks, the radiation pattern is not ideal. Also, the filler driver response has slopes of only 6 dB per octave, which may be insufficient. [2]

Much effort has been put into finding all-pass transfer functions for two-way systems. The Linkwitz-Riley crossover is one example [3]. Here the low-pass section has the same (even) order as the high-pass section, and the drivers are in phase at all frequencies. But with more than two-way, magnitude errors occur. [4]

Duelund transfer functions

Steen Duelund have describe an all-pass network with multiple in-phase drivers and slopes of a minimum of 12dB per octave, by combining a second order constant-voltage transfer function and a second order all-pass transfer function. [5]

$$DU4 := \frac{1 + a \cdot s + s^2}{1 + a \cdot s + s^2} \cdot \frac{1 - a \cdot s + s^2}{1 + a \cdot s + s^2} \qquad s := j \cdot \frac{f}{fc}$$

The expression is of the 4^{th} order. « j » is the complex operator « f » is the variable frequency, « fc » is the transitional frequency, « a » give the size of the damping.

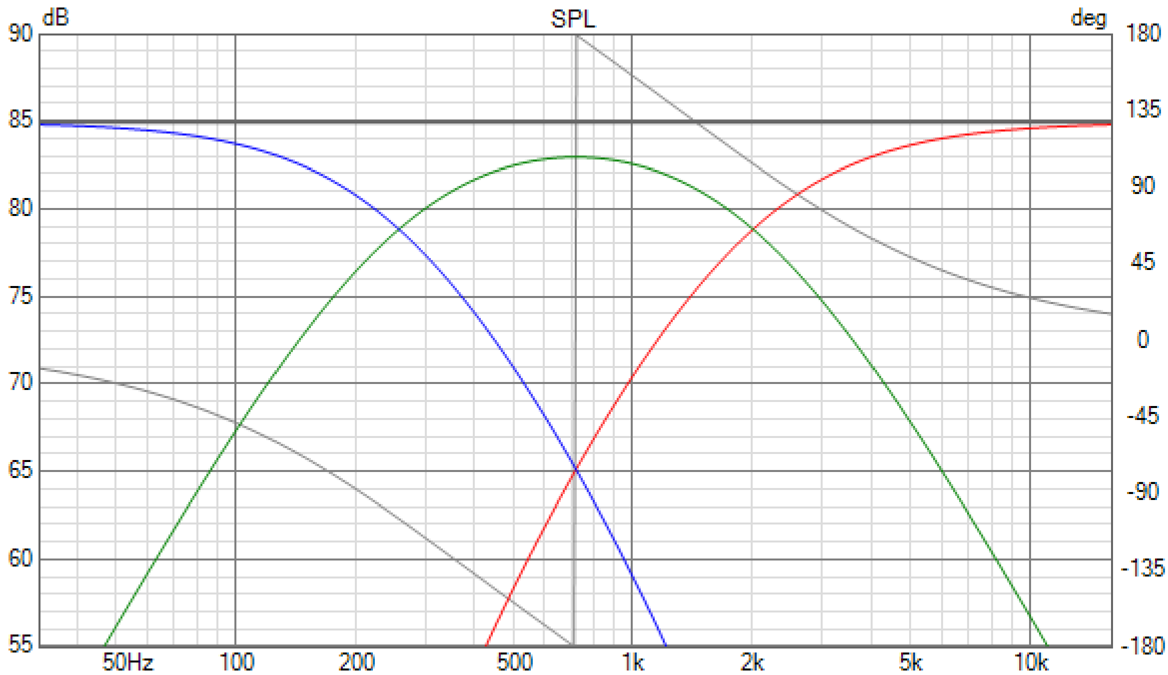
The development give low-pass, mid-pass and high-pass functions for three-way system :

$$Low\_pass\_DU4 := \frac{1}{\left(1 + a \cdot s + s^2\right)^2}$$

$$Mid\_pass\_DU4 := \frac{\left(2 - a^2\right) \cdot s^2}{\left(1 + a \cdot s + s^2\right)^2}$$

$$High\_pass\_DU4 := \frac{s^4}{\left(1 + a \cdot s + s^2\right)^2}$$

For example, with « fc » to 716 Hz and « a » to 3.13, the audio range is split with two crossovers, at 250Hz and 2kHz (offset at 85dB).



The sum have a constant amplitude response, and the four phases are synchronous (LP, MP, HP and Sum). The software used for this traces is VituixCAD [6] where each (pseudo-)drivers is associated with a .frd file generated by a Python script :

```
# Python script for "Duelund 4th 3 ways"

import math
import cmath

fc=716 # centre frequency
a=3.13 # factor "a" (damping)
offset=85 # targets dB offset
```

```
inf=15.625 # start frequency
sup=32000 # end frequency
N=264 # number of points (24 * 11 octaves = 264)

file_low=open("low_Duelund.frd","w") # names of files
file_mid=open("mid_Duelund.frd","w")
file_high=open("high_Duelund.frd","w")

for i in range(0,N+1):

f=inf*(sup/inf)**(i/N) # frequency sweep
s=1j*f/fc

low=1/(1+a*s+s**2)**2 # transfer functions
mid=(2-a**2)*low*s**2
high=low*s**4

file_low.write(str(f))
file_low.write("\t")
file_low.write(str(20*math.log10(abs(low))+offset))
file_low.write("\t")
file_low.write(str(cmath.phase(low)*180/math.pi))
file_low.write("\n")

file_mid.write(str(f))
file_mid.write("\t")
file_mid.write(str(20*math.log10(abs(mid))+offset))
file_mid.write("\t")
file_mid.write(str(cmath.phase(mid)*180/math.pi))
file_mid.write("\n")

file_high.write(str(f))
file_high.write("\t")
file_high.write(str(20*math.log10(abs(high))+offset))
file_high.write("\t")
file_high.write(str(cmath.phase(high)*180/math.pi))
file_high.write("\n")

file_low.close()
file_mid.close()
file_high.close()
```

Use of acoustic targets

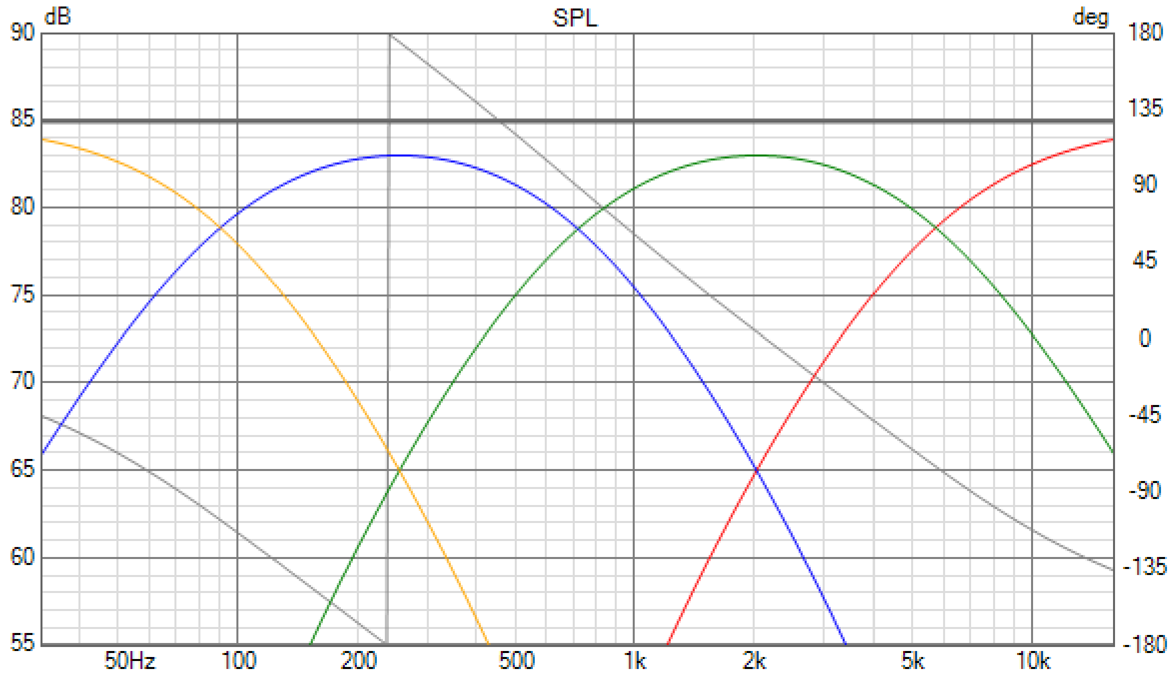
The acoustical output is of course dependent on both the electrical crossover network and the drivers, so the systems should ideally incorporate both contributions. [7]
Each driver must be measured without filter for implementation of real drivers in VituixCAD. With his tool "Optimizer", load each target file and build at the "crossover window", the electrical filter for join as best as possible the target.

Duelund functions with higher orders

$$DU6 := DU4 \cdot \frac{1+s}{1+s} \cdot \frac{1-s}{1+s}$$

```
low =       1/((1+2*s+s**2)*(1+a*s+s**2)**2)
mid_low =   low*(1-a**2)*s**2
mid_high =  low*(a**2-1)*s**4
high =      -low*s**6
```

Here with « fc » to 716 Hz and « a » to 8 :



$$DU8 := (DU4)^2$$

```
low =       1/(1+a*s+s**2)**4
mid_a =     low*(4-2*a**2)*s**2
mid_b =     low*(2+(2-a**2)**2)*s**4
mid_c =     low*(4-2*a**2)*s**6
high =      low*s**8
```

$$DU10 := DU8 \cdot \frac{1+s}{1+s} \cdot \frac{1-s}{1+s}$$

```
low =       1/((1+a*s+s**2)**4*(1+s)**2)
mid_a =     low*(3-2*a**2)*s**2
mid_b =     low*(2-2*a**2+a**4)*s**4
mid_c =     low*(-2+2*a**2-a**4)*s**6
mid_d =     low*(-3+2*a**2)*s**8
high =      -low*s**10
```

Conclusion

Parameters « fc », « a » and order in Duelund functions, offer a wide range of set of targets for multiway loudspeaker projects.

References

[1] René Christensen, "Active All-Pass Crossover Networks with Equal Resistors and Equal Capacitors". J. Audio Eng. Soc., Vol. 54, No. 1/2, 2006 February.

[2] Erik Baekgaard, "A Novel Approach to Linear Phase Loudspeakers Using Passive Crossover Networks" J. Audio Eng. Soc., vol. 25, pp. 284-294, 1977 May.

[3] Siegfried Linkwitz, "Active Crossover Networks for Noncoincident Drivers" J. Audio Eng. Soc., vol. 24, pp. 2-8, 1976 Jan./Feb.

[4] Siegfried Linkwitz,"Crossover topology issues" : https://www.linkwitzlab.com/frontiers_5.htm#V

[5] Steen Duelund, "The art of building a loudspeaker to the end " : http://www.steenduelund.dk/download/duelund-filter.pdf (Synkron filter)

[6] VituixCAD Loudspeaker simulator : https://kimmosaunisto.net/

[7] A Target Function approach to the design of filters : https://2adiy.ovh/KEFTOPICS_vol2no1.pdf