

2. Frequency Warping

January 25, 2017

1 Frequency Warping

These properties of the allpass can now be used to “warp” the frequency scale of a filter (by effectively replacing $e^j \leftarrow e^{j(\cdot)}$ in our frequency response), for instance to map it according to the so-called **Bark scale**, used in psycho-acoustics.

A common approximation of the Bark scale is ###

$$Bark = 13\arctan(0.0076f) + 3.5\arctan((f/7500)^2)$$

(From Wikipedia, Bark scale, the approximation goes back to Zwicker and Terhard), where f is the frequency in Hz. The Bark scale can be seen as an approximation of the changing frequency resolution over frequency of the inner ear filters of the human cochlea.

Because of the structure of our cochlea, the ear has different sensitivities for different frequencies and different signals. The signal dependent threshold of audibility of the ear is called the Masking Threshold. It has more spectral detail at lower than at higher frequencies, according to the Bark scale.

We can plot it using the python program

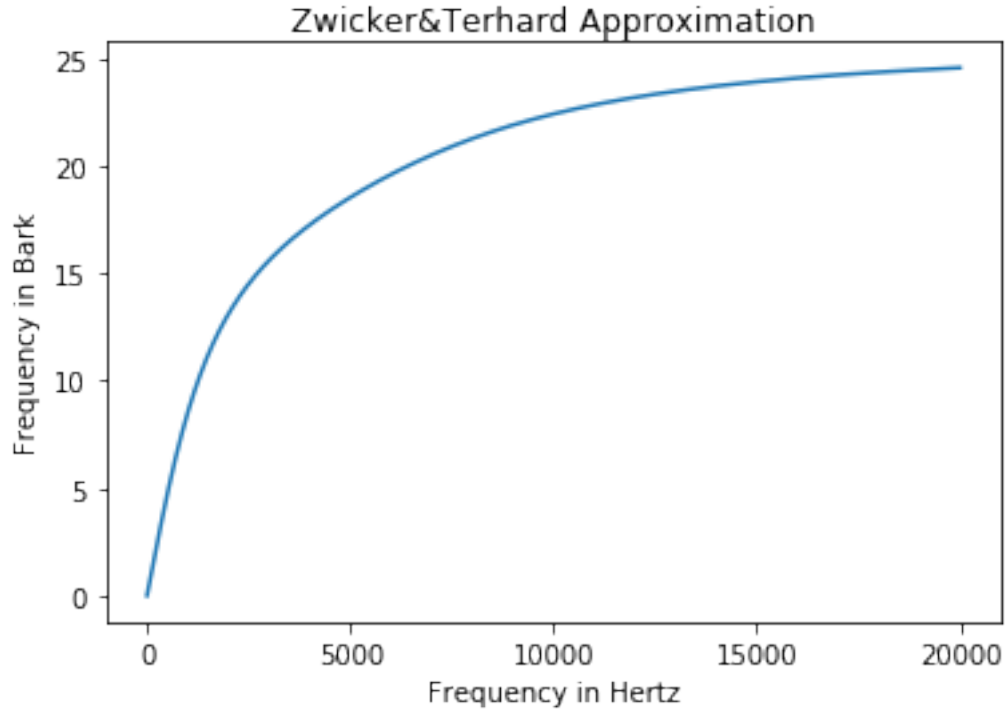
```
In [1]: %matplotlib inline
import numpy as np
import matplotlib.pyplot as plt

#Frequency array between 0 and 20000 Hz in 1000 steps:
f = np.linspace(0,20000,1000)

#Computation of Zwickers Bark approximation formula:
z = 13 * np.arctan(0.00076 * f) + 3.5 * np.arctan((f / 7500.0) ** 2)

#plot Bark over Hertz:
plt.plot(f,z)
plt.xlabel('Frequency in Hertz')
plt.ylabel('Frequency in Bark')
plt.title('Zwicker&Terhard Approximation')

Out[1]: <matplotlib.text.Text at 0x860f430>
```



Here we can see, that 1 bark at lower frequency has a much lower bandwidth than at higher frequencies. This means the ear can be seen as having a higher frequency resolution at lower frequencies than at higher frequencies. Imagine, we want to **design a filter** or system for **hearing purposes**, for instance, we would like to model the masking threshold of the ear for any given signal by some linear filter (FIR or IIR). Then it would be useful, to give this filter a **higher frequency resolution at lower frequencies**, such that it matches the smaller details of the masking threshold at lower frequencies. But if we look at **the usual design methods**, they distribute the filter details **independent of the frequency range** (for instance what we saw with the remez method, where we have equally distributed ripples). Here we can now use frequency warping, such that **we enlarge the low frequency range** and shrink the high frequency range accordingly, such that our filter now works on the **warped frequency**, and **“sees” the lower frequencies in more detail**, the lower frequencies are more spread out in comparison to the higher frequencies.

How do we do this? For some frequency response $H(e^j)$ we would like to warp the frequency with some function $()$ according to our desired frequency scale, such that we get ###

$$H(e^{j\omega})$$

. But this is exactly the principle of an **allpass filter**, which has the frequency response ###

$$H_{ap}(e^j) = e^{j\omega_p()}$$

Usually we would like to map positive frequencies to again positive frequencies, and we saw that $\omega_p()$ becomes negative, hence we take the approach to **replace z** in the argument of our transfer function with the reverse of our **allpass** transfer function: ###

$$z^{-1} \leftarrow H_{ap}(a, z)$$

This is replacing all delays of our filter to be warped by our allpass filter. In this way we replace our linear function on the unit circle in z with the non-linear, warped function on the unit circle H_{ap} .

Hence we get the warped transfer function as ###

$$H_{warped}(z) = H(H_{ap}(a, z)^1)$$

and the resulting frequency response becomes ###

$$H_{warped}(e^j) = H(e^{j_{ap}()})$$

Here we can now see that we obtained the **desired frequency warping**.