Albert Iepure
Matrikelnummer: 03650065

Chromatic dispersion compensation using complex-valued all-pass filter

In this project the principle of Chromatic Dispersion (CD) in optical channel is described, focusing on the part that it has an all-pass behavior, which means that the amplitude of the signal that passes through the channel is left unchanged, whereas the phase of the signal is changed due to CD. This CD can be compensated by using IIR or FIR filters. In this project I implemented an IIR compensation filter with complex coefficients for compensating CD.

3. Practical examples

Abel-Smith algorithm

The Abel-Smith algorithm is a method to extract the filter coefficients for an all-pass design with an arbitrary group delay. The main principle is to divide the area below $\tau_{desired}(\omega)$ into 2π area frequency bands. The area below $\tau_{desired}(\omega)$ is guaranteed to be an integer multiple of 2π because it is equal to $2\pi N_{IIR}$.

With the given parameters a value for alpha equal to α=0.7360 is obtained and a value for N_{IIR} = 5. This two parameters are input parameters for the *abel_smith* function, which returns the ρ and θ vectors. In this function, I chose ζ=0.8 (at the middle of the interval [0.75, 0.85]), after which the function *abel_smith_divide* is called, which takes as input parameters also α and N_{IIR} and outputs the vector called *omega_seg* which contains the pulsations between $[-\pi,\pi]$ for which the area under $\tau_{desired}(\omega)$ is 2π.

Within this function, in the first step β is computed and is equal to $\beta = N_{IIR} = 5$. Afterwards, the first and last value of omega_seg are initialized with $-\pi$ and π respectively (omega_seg has N_{IIR} elements). In order to find the other values of ω in-between I did the following: I took the equation of the line describing $\tau_{desired}(\omega)$ and I chose two points on it: at the first iteration I chose $\omega_0=-\pi$ and ω_1 which is to be determined. The two points, together with the OX axis describe a trapezoid, whose area can be expressed in terms of ω_1 and ω_0, and this area is also equal to 2π. Doing this, we obtain a second order equation, which yields two solutions. From these two solutions, we choose only the one who is smaller in absolute value than the one previously determined. The algorithm is repeated until all points are calculated. A detailed derivation of the equation is included In the Appendix, page 3.

The values obtained for omega-seg are:

[-3.1415926535,-2.4524196594,-1.6703670709, -0.7431304218,0.4638963171,3.1415926535]

These values are then used to compute the ρ and θ vectors according to the formulas:

Albert Iepure
Matrikelnummer: 03650065

The pole frequency is taken to be the band midpoint,

$$\theta_i = \frac{\omega_{i-1} + \omega_i}{2}.$$

The expression for pole radius is derived as,

$$\rho_i = \mu_i - \sqrt{\mu_i^2 - 1}\ ,$$

where

$$\mu_i = \frac{1 - \zeta \cdot \cos(\Delta_i)}{1 - \zeta}, \qquad \Delta_i = \frac{\omega_i - \omega_{i-1}}{2},$$

At the end, the values for ρ and θ are:

ρ = [0.5101779789,0.4682523645,0.4109212203,0.3236712115,0.1244310801]

θ = [-2.7970061565,-2.0613933652,-1.2067487464,-0.1396170523,1.8027444853]

Bit error rate

In this exercise a CD channel and CD equalization are implemented and a BER curve is obtained.

In the first task I generate the channel impulse response. For this I created a function called impulse_response_channel which takes as inputs α and the number of frequency points and outputs h_CD. Within this function, after computing the omega values in the interval [-π,π] (namely the frequency points where the transfer function is going to be estimated), I compute:

```
H_CD = exp(-1i*alpha*omega.^2)
```

After this, I apply the ***ifftshift*** to H_CD because this function of Matlab works on the domain from [0,2π], and the frequency range for H_CD is [-π,π]. Thus the domain between [0, π] coincides, whereas the interval from [π,2π] has to be shifted at the beginning of the H_CD vector. This being done, then the impulse response h_CD is computed by taking the ***IFFT*** of H_CD. After assuring that h_CD is a column vector (through the command `h_CD = h_CD(:)`), the command ***fftshift*** is applied to h_CD for a similar reason with the one described above. Finally,the channel response is plotted by making use of the ***stem*** command. The result can be seen Figure 1.

After loading the values for ρ,θ and Φ another function is called, namely *conv_anyinput_allpass_equalizer* which takes as input arguments the previous variables, together with h_CD and outputs the result of the convolution between the IIR filter described by ρ,θ,Φ and the input signal. Within this function the ***filter*** function of Matlab is used, together with the cascading of N_{IIR} IIR order 1 filters, for which the corresponding b and a vectors are shown below:

```
for k=1:length(rho)
b = [-rho(k)*exp(-1i*theta(k)), 1];
a = [1, -rho(k)*exp(1i*theta(k))];
output = filter(b,a,input);
input = output;
end

output = exp(-1i*phi).*output;
```

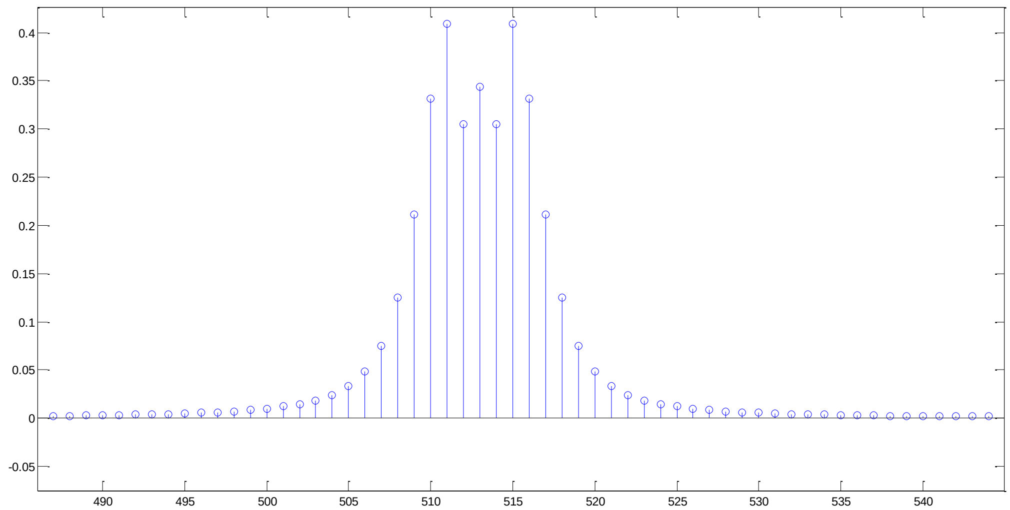


Figure 1: Impulse response of the CD channel

After computing the output of one such filter, this signal becomes the input for the following order 1 IIR filter, because they are cascaded. In the end, the output is multiplied by $e^{-j\Phi}$ to obtain the desired phase response. The output of this function is shown in Figure 2.

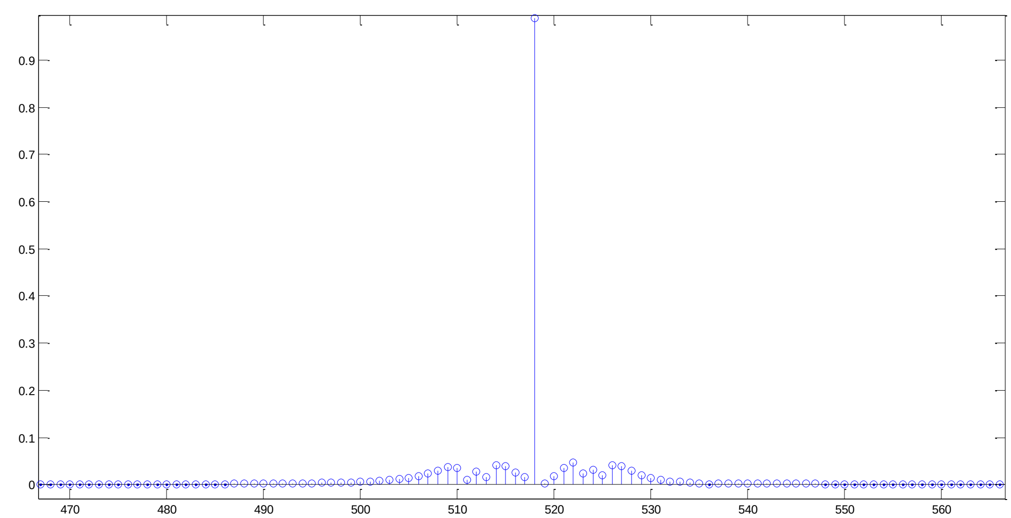
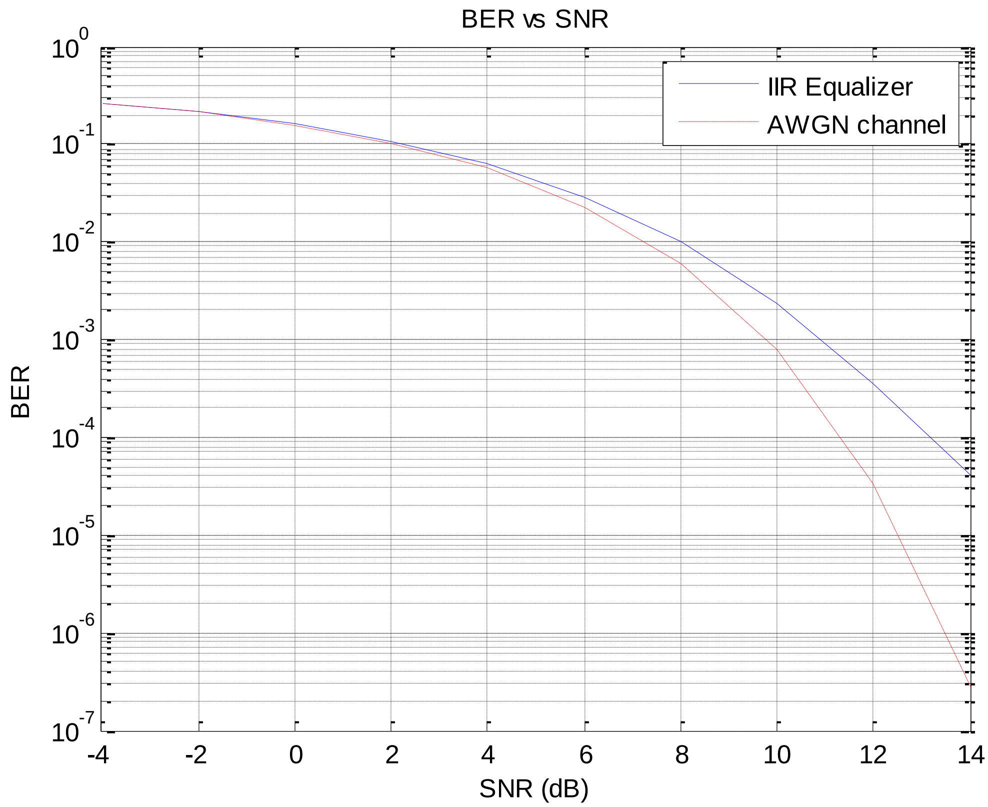


Figure 2: Output after CD equalization

As it can be easily determined from Figure 2, the output after the equalization is nearly a Dirac impulse of amplitude 0.9892 at sample number 518.

After this the BER vs. SNR curve is plotted and is shown in Figure 3.



Albert Iepure
Matrikelnummer: 03650065

c) Frequency sampling

In the frequency sampling exercise we are required to compute the channel response for the first 256 samples.

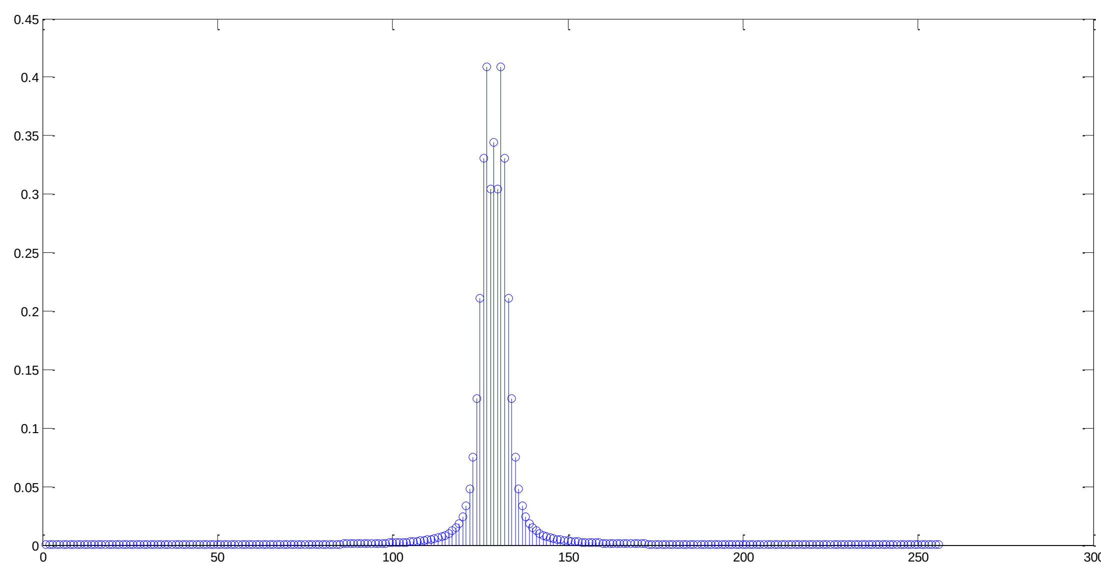
Given the input values, I obtained α=0.7355. Then I created a function called impulse_response_channel which takes as input arguments α and the number of equally spaced frequency points in the interval $[-\pi,\pi]$. In the function, after computing the omega values (namely the frequency points where the transfer function is going to be estimated), I compute

```
H_CD = exp(-1i*alpha*omega.^2)
```

After this, I apply the ***ifftshift*** to H_CD because this function of Matlab works on the domain from $[0,2\pi]$, and my frequency range for H_CD is $[-\pi,\pi]$. Thus the domain between $[0, \pi]$ coincides, whereas the interval from $[\pi,2\pi]$ has to be shifted at the beginning of the H_CD vector.

This being done, I then compute the impulse response h_CD by taking the ***IFFT*** of H_CD. After assuring that h_CD is a column vector (through the command `h_CD = h_CD(:)`), I use the command ***fftshift*** applied to h_CD for a similar reason with the one described above.

Finally, I plot the channel response of the channel by making use of the ***stem*** command. The result can be seen below.



Channel response of the channel for the first 256 samples

Albert
Tepure
03650065

Project 2: Chromatic dispersion compensation using complex-valued all-pass filters
– Appendix –

2nd Task: Preparation exercises

a) Discretisation

$$H_{CS}(\Omega) = \exp\left(-j\frac{\lambda_0^2}{4\pi c}\Delta L \Omega^2\right)$$

Ω = baseband radial frequency $= 2\pi f$

$\omega = \frac{\Omega}{B}$ normalised frequency $\Rightarrow \Omega = \omega B$

$$H_{CS}(\omega) = \exp\left(-j\frac{\lambda_0^2}{4\pi c}\Delta L B^2 \omega^2\right), \text{ where } \alpha = \frac{\lambda_0^2}{4\pi c}\Delta L B^2 \quad \text{q.e.d.}$$

b) Ideal equalizer

We want that $H_{CS}(\omega) \cdot G_{ideal}(\omega) = 1 \Rightarrow G_{ideal}(\omega) = \frac{1}{H_{CS}(\omega)} = e^{j\alpha\omega^2}$ q.e.d

c) Frequency sampling

See page 5 of the Appendix

d) Number of taps

$$N_{FIR} \simeq 2\left\lfloor \frac{\lambda_0^2}{2c}\Delta B^2 L \right\rfloor + 1 = 2\left\lfloor \frac{1550^2 \cdot \cancel{10^{-18}}}{2\cdot 3\cdot 10^8} \cdot 16 \cdot \frac{\cancel{10^{-12}}\, 10^{-3}}{\cancel{10^{-9}}\cdot\cancel{10^{3}}} \cdot 56 \cdot \cancel{10^{18}} \cdot 23 \cdot \cancel{10^{3}} \right\rfloor + 1$$

$$= 2\lfloor 4.621 \rfloor + 1 = 2\cdot 4 + 1 = 9$$

So, for the equalisation with an FIR filter $N_{FIR}+1 = 10$ taps are necessary, because the number of taps is given by the number of coefficients of the impulse response, which is the number of delay elements plus one.

e) Group delay

$$\tau_{IIR}(\omega) = -\frac{\partial}{\partial\omega}\phi_{IIR}(\omega)$$

$$\phi_{IIR}(\omega) = \sum_{i=1}^{N_{IIR}}\left[-\omega - 2\arctan\left(\frac{\rho_i \sin(\omega-\theta_i)}{1-\rho_i\cos(\omega-\theta_i)}\right)\right]$$

Using $\frac{d}{dx}\arctan x = \frac{1}{1+x^2}$ we get:

$\sin' x = \cos x$

$\cos' x = -\sin x$

$$\tau_{iiR}(\omega) = -\frac{\partial \phi_{IIR}}{\partial \omega} = -\sum_{i=1}^{N_{IIR}} \left[-1 - \frac{2}{\frac{1+\rho_i^2 \sin^2(\omega-\theta_i)}{[1-\rho_i\cos(\omega-\theta_i)]^2}} \cdot \frac{\rho_i \cos(\omega-\theta_i)(1-\rho_i\cos(\omega-\theta_i)) - \rho_i \sin(\omega-\theta_i)\rho_i \sin(\omega-\theta_i)}{(1-\rho_i\cos(\omega-\theta_i))^2} \right]$$

$$= \sum_{i=1}^{N_{IIR}} 1 + \frac{2\cancel{[1-\rho_i\cos(\omega-\theta_i)]^2}[\rho_i\cos(\omega-\theta_i) - \rho_i^2\cos^2(\omega-\theta_i) - \rho_i^2\sin^2(\omega-\theta_i)]}{[1-2\rho_i\cos(\omega-\theta_i)+\rho_i^2\cos^2(\omega-\theta_i)+\rho_i^2\sin^2(\omega-\theta_i)]\cancel{[1-\rho_i\cos(\omega-\theta_i)]^2}}$$

$$= \sum_{i=1}^{N_{IIR}} 1 + \frac{2[\rho_i\cos(\omega-\theta_i) - \rho_i^2]}{1-2\rho_i\cos(\omega-\theta_i)+\rho_i^2} =$$

$$= \sum_{i=1}^{N_{IIR}} \frac{1 \cancel{- 2\rho_i\cos(\omega-\theta_i)} + \rho_i^2 \cancel{+ 2\rho_i\cos(\omega-\theta_i)} - 2\rho_i^2}{1-2\rho_i\cos(\omega-\theta_i)+\rho_i^2} = \sum_{i=1}^{N_{IIR}} \frac{1-\rho_i^2}{1-2\rho_i\cos(\omega-\theta_i)+\rho_i^2}$$

q.e.d.

f) Group delay extrema

$$N_{IIR} = 1 \Rightarrow \tau_{iiR}(\omega) = \frac{1-\rho_1^2}{1-2\rho_1\cos(\omega-\theta_1)+\rho_1^2}$$

$$\frac{\partial \tau_{iiR}(\omega)}{\partial \rho_1} = \frac{-2\rho_1(1-2\rho_1\cos(\omega-\theta_1)+\rho_1^2) - (1-\rho_1^2)(-2\cos(\omega-\theta_1)+2\rho_1)}{[1-2\rho_1\cos(\omega-\theta_1)+\rho_1^2]^2} \overset{!}{=} 0$$

$$-2\rho_1 + 4\rho_1^2\cos(\omega-\theta_1) \cancel{-2\rho_1^3} + 2\cos(\omega-\theta_1) - 2\rho_1 - 2\rho_1^2\cos(\omega-\theta_1) \cancel{+2\rho_1^3} = 0$$

$$2\rho_1^2\cos(\omega-\theta_1) - 4\rho_1 + 2\cos(\omega-\theta_1) = 0 \Rightarrow \rho_1^2\cos(\omega-\theta_1) - 2\rho_1 + \cos(\omega-\theta_1) = 0$$

$$\Delta = 4 - 4\cos^2(\omega-\theta_1);\ \sqrt{\Delta} = 2\sqrt{1-\cos^2(\omega-\theta_1)} = 2\sqrt{\cancel{\cos^2(\omega-\theta_1)} + \sin^2(\omega-\theta_1) \cancel{- \cos^2(\omega-\theta_1)}}$$

$$\sqrt{\Delta} = 2\sin(\omega-\theta_1)$$

$$\rho = \frac{2 \pm 2\sin(\omega-\theta_1)}{2\cos(\omega-\theta_1)} = \frac{1 \pm \sin(\omega-\theta_1)}{\cos(\omega-\theta_1)}$$

$$\rho_1 = \frac{1+\sin(\omega-\theta_1)}{\cos(\omega-\theta_1)} \Rightarrow \tau_{iiR}(\omega) = \frac{1 - \frac{1+2\sin(\omega-\theta_1)+\sin^2(\omega-\theta_1)}{\cos^2(\omega-\theta_1)}}{1 - 2\frac{[1+\sin(\omega-\theta_1)]}{\cos(\omega-\theta_1)}\cdot\cos(\omega-\theta_1) + \frac{1+2\sin(\omega-\theta_1)+\sin^2(\omega-\theta_1)}{\cos^2(\omega-\theta_1)}}$$

...

So we observe that we have $\cos(\omega-\theta_1)$ at the denominator $\Rightarrow$ the minimum/maximum values of the denominator are achieved when $\cos(\omega-\theta_1) = \pm 1$, i.e. $1-2\rho_1+\rho_1^2 = (1-\rho_1)^2$ and $1+2\rho_1+\rho_1^2 = (1+\rho_1)^2$. Thus, the denominator is always positive.

Now we can distinguish two cases for ρ_1:

1° $0 < \rho_1 < 1 \Rightarrow 1-\rho_1^2 > 0 \Rightarrow$
(this means the IIR filter is stable)

$$\begin{cases} \tau_{iiR,min}(\omega) = \frac{1-\rho_1^2}{(1+\rho_1)^2} = \frac{1-\rho_1}{1+\rho_1} \\ \tau_{iiR,max}(\omega) = \frac{1-\rho_1^2}{(1-\rho_1)^2} = \frac{\cancel{(1-\rho_1)}(1+\rho_1)}{(1-\rho_1)^{\cancel{2}}} = \frac{1+\rho_1}{1-\rho_1} \end{cases}$$

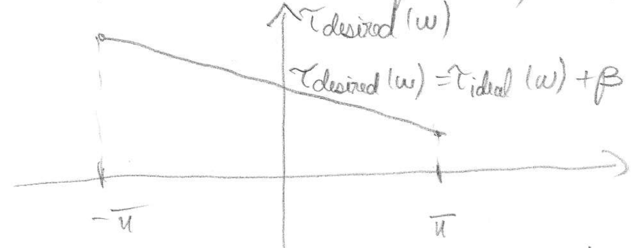
2° $1 \le \rho_1 \Rightarrow 1-\rho_1^2 \le 0 \Rightarrow$
(this means the IIR filter is unstable)

$$\begin{cases} \tau_{iiR,min}(\omega) = \frac{1-\rho_1^2}{(1-\rho_1)^2} = \frac{1+\rho_1}{1-\rho_1} \\ \tau_{iiR,max}(\omega) = \frac{1-\rho_1^2}{(1+\rho_1)^2} = \frac{1-\rho_1}{1+\rho_1} \end{cases}$$

g) Integer factor

We have $\tau_{desired}(w) = -2\alpha w + \beta$

We want this to be positive, therefore we have the graph as shown below



We observe that the min value of $\tau_{desired}(w)$ is obtained for $w = \pi$

$\Rightarrow -2\alpha\pi + \beta > 0 \Rightarrow \beta > 2\alpha\pi$

We take the nearest integer closest to this value $\Rightarrow \beta = \lceil 2\alpha\pi \rceil$

h) Desired phase response

$\tau_{desired}(w) = -2\alpha w + \beta$

The group delay is obtained by taking the negative gradient of the phase response with respect to w $\Rightarrow$ the desired phase response is obtained by integrating the desired group delay multiplied by -1:

$$\phi_{desired}(w) = \int (2\alpha w - \beta)\, dw = \frac{2\alpha}{2}\int (w^2)'\, dw - \beta \int w'\, dw =$$

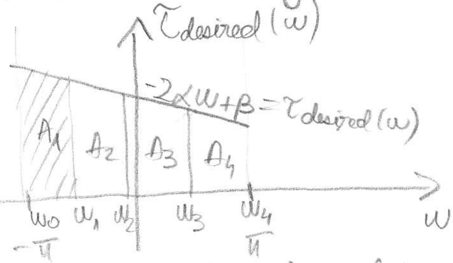
$$= \alpha w^2 - \beta w + \underbrace{C}_{\text{integration constant}}$$

By denoting the integration constant C by $-\phi_0$ we obtain:

$\phi_{desired}(w) = \alpha w^2 - \beta w - \phi_0$ q.e.d.

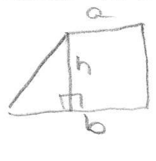
3. Practical examples

Abel-Smith algorithm



$$A_1 = \frac{[\tau_{desired}(w_0) + \tau_{desired}(w_1)](w_1 - w_0)}{2}$$

because the area of a trapezoid is:



$$A = \frac{(a+b)h}{2}$$

We want $A_1 = A_2 = A_3 = A_4 = 2\pi \Rightarrow$

$\Rightarrow 4\pi = (-2\alpha w_0 + \beta - 2\alpha w_1 + \beta)(w_1 - w_0) \Rightarrow 4\pi = [-2\alpha(w_0 + w_1) + 2\beta](w_1 - w_0)$

$\Rightarrow 4\pi = -2\alpha(w_1^2 - w_0^2) + 2\beta(w_1 - w_0) \Rightarrow 2\alpha(w_1^2 - w_0^2) - 2\beta(w_1 - w_0) + 4\pi = 0$

$\Rightarrow 2\alpha w_1^2 - 2\beta w_1 + (4\pi - 2\alpha w_0^2 + 2\beta w_0) = 0$

$\Delta = 4\beta^2 - 8\alpha(4\pi - 2\alpha w_0^2 + 2\beta w_0)$

$w_{1,2} = \frac{2\beta \pm \sqrt{\Delta}}{4\alpha}$ (w_0 is already known) $\Bigg| \Rightarrow \underline{w_1}$

Condition: $|w_1| < |w_0|$

The algorithm repeats itself until all w_i, $i = 1 : N_{IIR}$ are calculated