Einf. in die digitale Signalverarbeitung - Übung I

Felix Rohrer

>

Aufgabe 1: Schallpegel

Wieviel Dezibel entsprechen schwachem Blätterrauschen bzw. einem startenden Flugzeug? Welche mathematische Beziehung besteht zwischen den Schallintensitäten dieser beiden Signale (Formel angeben)?

> *restart*; *Digits* := 4 :

> *schwBlaetterrauschen* := 10 *#db, schwaches Blätterrauschen*

$$schwBlaetterrauschen := 10 \tag{1.1}$$

> *Flugzeug* := 120 *#db Flugzeug bei 100m Distanz*

$$Flugzeug := 120 \tag{1.2}$$

Schallpegel vs. Schallintensität

$$\mathbf{120 - 10 = 110 = 10 \cdot \log_{10}\left(\frac{I_B}{I_F}\right)}$$

>

Aufgabe 2: Diskretisierung/Quantisierung

Diskretisieren Sie das Sinus-Signal (1) der Frequenz $f = 440\,\mathrm{Hz}$ mit der Abtastfrequenz $f_a = 8192\,\mathrm{Hz}$, wobei Sie 8 Bit (d.h. Werte von -127 bis 127) verwenden. Was ändert sich qualitativ, wenn Sie statt linearer Kodierung die μ-Law Kodierung verwenden?

$$y(t) = \sin(2\pi f\, t), \quad t \in [0, 2\pi) \tag{1}$$

> *restart* : *Digits* := 4

$$Digits := 4 \tag{2.1}$$

> $y := t \rightarrow \sin(2 \cdot \pi \cdot f \cdot t)$

$$y := t \rightarrow \sin(2\,\pi\, f\, t) \tag{2.2}$$

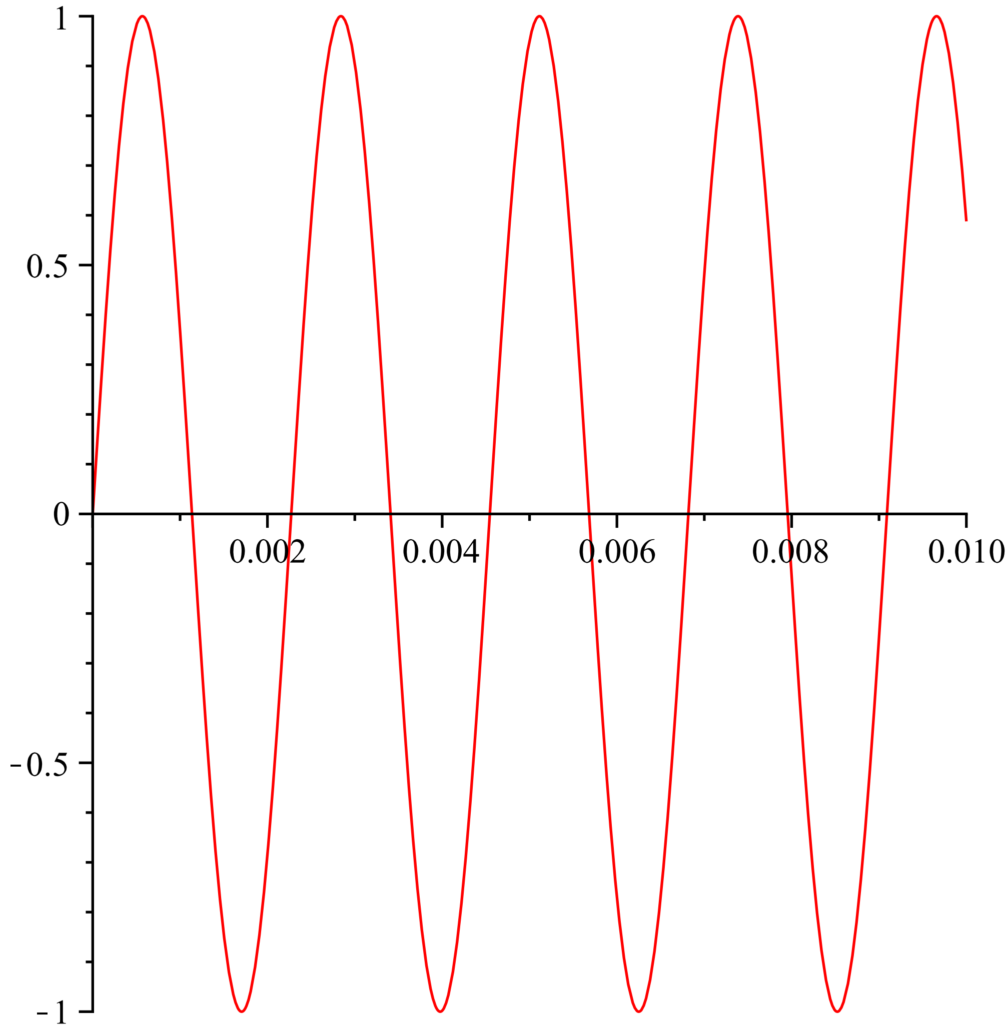
> $f := 440$

$$f := 440 \tag{2.3}$$

> $fa := 8192$

$$fa := 8192 \tag{2.4}$$

> *plot*(*y*, 0 ..0.01)



> **for** i **from** 1 **to** 12 **do** $evalf\left(y\left(\frac{i}{fa}\right)\cdot 127+1\right)$ **do**

$$43.05$$
$$80.35$$
$$108.7$$
$$124.9$$
$$127.1$$
$$115.1$$
$$90.24$$
$$55.31$$
$$14.22$$
$$-28.35$$
$$-67.59$$
$$-99.1 \quad \textbf{(2.5)}$$

Bei µLaw wird nicht linear kodiert. Für leise Töne werden mehr Bits verwendet (in diesem Range) also für den restlichen Range (laute Töne).

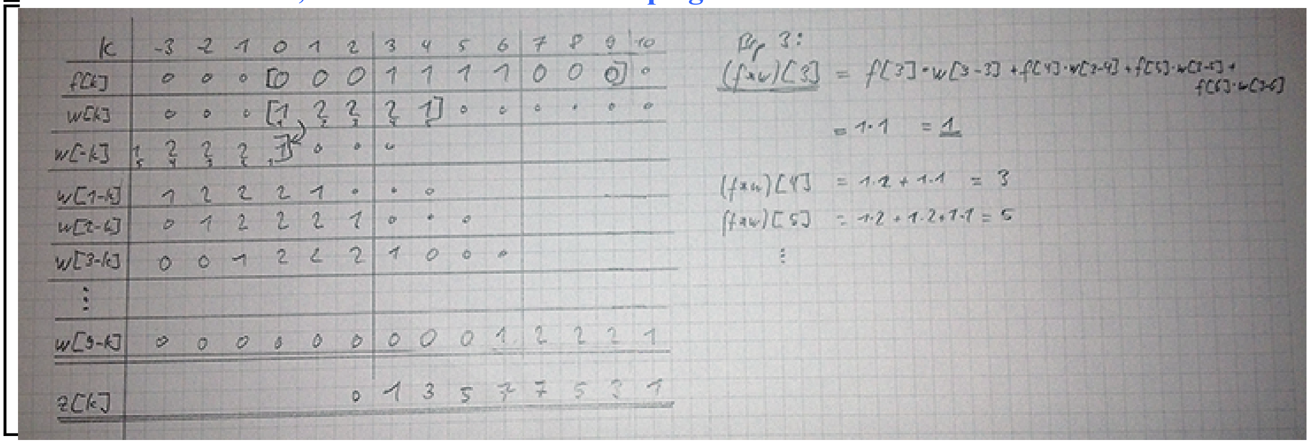
>

Aufgabe 3: Faltung

Ein eindimensionales Bild enthalte die folgenden Graustufenwerte $f = \{0, 0, 0, 1, 1, 1, 1, 0, 0, 0\}$. Es soll mit der Maske $w = \{1, 2, 2, 2, 1\}$ gefaltet werden.

Berechnen Sie die Faltung $(f \star w)$ von Hand und zur Kontrolle mit Hilfe geeigneter Matlab-Befehle. Auf was ist bei der Berechnung zu achten?

Mit Nullen auffüllen, bei Position 0 Maske w spiegeln!



k	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10
f[k]	0	0	0	[0	0	0	1	1	1	1	0	0	0]	0
w[k]	0	0	0	[1	2	2	2	1]	0	0	0	0	0	0
w[-k] 1	2	2	2	1	0	0	0							
w[1-k]	1	2	2	2	1	0	0	0						
w[2-k]	0	1	2	2	2	1	0	0	0					
w[3-k]	0	0	1	2	2	2	1	0	0	0				
⋮														
w[9-k]	0	0	0	0	0	0	0	0	0	1	2	2	2	1
z[k]						0	1	3	5	7	7	5	3	1

Bsp 3:

$(f*w)[3] = f[3]\cdot w[3-3] + f[4]\cdot w[3-4] + f[5]\cdot w[3-5] + f[6]\cdot w[3-6]$

$= 1\cdot 1 = \underline{1}$

$(f*w)[4] = 1\cdot 2 + 1\cdot 1 = 3$

$(f*w)[5] = 1\cdot 2 + 1\cdot 2 + 1\cdot 1 = 5$

⋮

Mit MatLab:

```
>> f = [ 0 0 0 1 1 1 1 0 0 0 ]

f =

     0     0     0     1     1     1     1     0     0     0

>> w = [ 1 2 2 2 1 ]

w =

     1     2     2     2     1

>> res = conv(f, w)

res =

     0     0     0     1     3     5     7     7     5     3     1     0     0     0
```

>

Aufgabe 4: Faltung und Barker-Codes

Falten Sie den Barker-Code der Länge 4 ($\{x_1[n]\} = \{1, 1, -1, 1\}$) mit der Folge $\{x_2[n]\} = \{1, -1, 1, 1\}$. Wo hat die Faltung ihr Maximum? Man kann zeigen: faltet man eine Barker-Codefolge mit ihrer zeitumgekehrten Folge, entsteht ein sehr eindeutiges Maximum bei der Länge der Barker-Codefolge.

Mit MatLab:

```
>> x1 = [1 1 -1 1]

x1 =

     1     1    -1     1

>> x2 = [1 -1 1 1]

x2 =

     1    -1     1     1

>> res = conv(x1, x2)

res =

     1     0    -1     4    -1     0     1
```

Bei der länge (in diesem Bsp. 4 -> Position 3) vom Barker-Code entsteht das Maximum.

>