

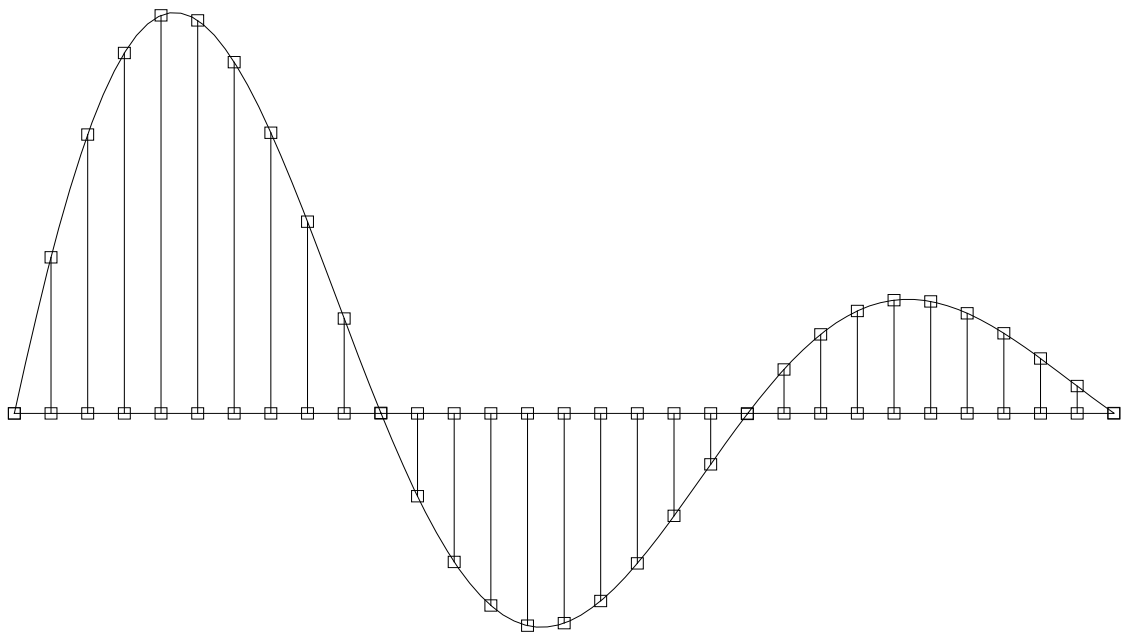
Aufgabensammlung zur Digitalen Signalverarbeitung

zusammengestellt von Gunnar Jahnke

11. Januar 1999

Zusammenfassung

Musterlösungen zur Vorlesung
„Digitale Signalverarbeitung in der Seismologie I und II“
(WS 95/96 und SS 96)
Basierend auf dem Skript „Digitale Signalverarbeitung“
von Gerhard Müller.



Inhaltsverzeichnis

1	FFT von verschiedenen Sinusimpulsen	3
1.1	Aufgabenstellung	3
1.2	Bemerkungen zu C Funktion <i>four1()</i>	3
1.3	Plots für Aufgabe 1	5
1.4	Listen der FT's	5
2	FFT von weiteren Impulsen	7
2.1	Aufgabenstellung	7
2.2	Listen der FT's	7
2.3	Bestimmung von $\Delta T \cdot \Delta f$	8
2.4	Plots für Aufgabe 2	9
3	Synthetisches Seismogramm	9
3.1	Aufgabenstellung	9
3.2	Plots für Aufgabe 3	10
3.3	Überlegungen zum Seismogramm	13
4	Faltung	13
4.1	Aufgabenstellung	13
4.2	Plots für Aufgabe 4	14
5	Aliasing	15
5.1	Aufgabenstellung	15
5.2	Plots für Aufgabe 5	15
6	Tiefpaßfilterung	17
6.1	Aufgabenstellung:	17
6.2	Erläuterungen	18
6.3	Plots für Aufgabe 6:	19
7	Bandpaßfilterung	25
7.1	Aufgabenstellung:	25
7.2	Plots für Aufgabe 7	25
7.3	Akausalität	26
8	Butterworth-Tiefpaß n-ter Ordnung	26
8.1	Aufgabenstellung:	26
8.2	Plots für Aufgabe 8:	27
9	Kreuzkorrelation	31
9.1	Aufgabenstellung:	31
9.2	Plots für Aufgabe 9:	32

10 Vibroseis	33
10.1 Aufgabenstellung:	33
10.2 Plots für Aufgabe 10:	34
11 Rekursives Notch-Filter	37
11.1 Aufgabenstellung:	37
11.2 Plots für Aufgabe 11:	38
12 Rekursiver Butterworth-Tiefpaß	40
12.1 Aufgabenstellung:	40
12.2 Plots für Aufgabe 12:	41
13 Analyse und Invertierung eines Wavelets	41
13.1 Aufgabenstellung:	41
13.2 Analytische Rechnungen	41
13.3 Listen und Fehlerenergien	43
13.4 Plots für Aufgabe 13:	44
14 Wiener Optimum Filter und Spike- Dekonvolution	46
14.1 Aufgabenstellung:	46
14.2 Bemerkungen zum Levinson-Algorithmus	46
14.3 Test mit dem Beispiel aus Abschnitt 10.1.2	47
14.4 Plots für Aufgabe 14:	47
15 Wiener Filter zur Unterdrückung von Rauschen	54
15.1 Aufgabenstellung:	54
15.2 Plots für Aufgabe 15:	55
16 Farbiges Rauschen / prädikative Dekonvolution	61
16.1 Aufgabenstellung:	61
16.2 Plots für Aufgabe 16:	61
17 Zu den Programmen	66
17.1 Struktur des Verzeichnisbaumes	66
17.2 Fast Fourier Transformation (FFT)	66
17.3 Levinson Algorithmus	67

1 FFT von verschiedenen Sinusimpulsen

Fourier-Transformation von Impulsen mit der FFT, Vergleich mit analytischen Resultaten

1.1 Aufgabenstellung

Man schreibe ein FORTRAN / C Programm, das folgendes durchführt:

a) Der Impuls

$$\begin{aligned} x(t) &= \sin\left(\frac{2\pi}{T_0} \cdot t\right) && \text{für } 0 \leq t \leq T_0 \\ x(t) &= 0 && \text{sonst} \end{aligned}$$

soll zwischen $t = 0$ und $t = T_1 \geq T_0$ berechnet, gelistet und geplottet werden.

Man setze: $T_1 = LX \cdot \Delta t$, $T_0 = LT \cdot \Delta t$

mit dem Zeitintervall Δt .

Zahlenwerte: $LX = 64$; $LT = 10, 20, 64$ $\Delta t = 0.002s$

b) Berechnung der FT (Fourier-Transformierten) von $x(t)$ mit *FORK* (FORTRAN) bzw. *four1()* (in C)

Liste von Frequenz, Real- und Imaginärteil der FT, Plot des Betrags.

c) Man berechne die FT von $x(t)$ analytisch (vgl. S 13/14 im Skript) und vergleiche mit den numerischen Resultaten (Nullstellen, Wert für die Hauptfrequenz $1/T_0$).

1.2 Bemerkungen zu C Funktion *four1()*

Fast Fourier Transformation (FFT) in C

Die Funktion *four1()* steht in den Numerical Recipies. Der Quelltext steht bei den programmierten Lösungen im File **four1.c**

Parameter in *four1()*

void four(float *daten,int wertezahl,int richtung)

daten : Feld von 'floats' für das (komplexe) Signal

wertezahl : Anzahl der Daten, die in dem Feld 'daten' stehen

richtung : -1 für Transformation Zeit- \Rightarrow Frequenzbereich
+1 für Frequenz- \Rightarrow Zeitbereich

Anordnung der Daten für *four1()*

Obwohl Felder bzw. Vektoren in C mit dem Index 0 anfangen, erwartet *four1()* ein Feld mit den Daten ab Index 1 in der Form:

Feldindex:	0	1	2	3	4	5	6	...
Inhalt:		R_1	I_1	R_2	I_2	R_3	I_3	...

mit: R_n : Realteil des n-ten Wertes
 I_n : Imaginärteil des n-ten Wertes

Also: Das erste Element bleibt frei, dann kommen abwechselnd Real- und Imaginärteile der Daten. Für reelle Signale sind die Imaginärteile Null zu setzen.

Möchte man die FFT von z.B. $n = 1024$ Daten berechnen, muß das Feld $2 \cdot n + 1 = 2049$ Einträge groß sein.

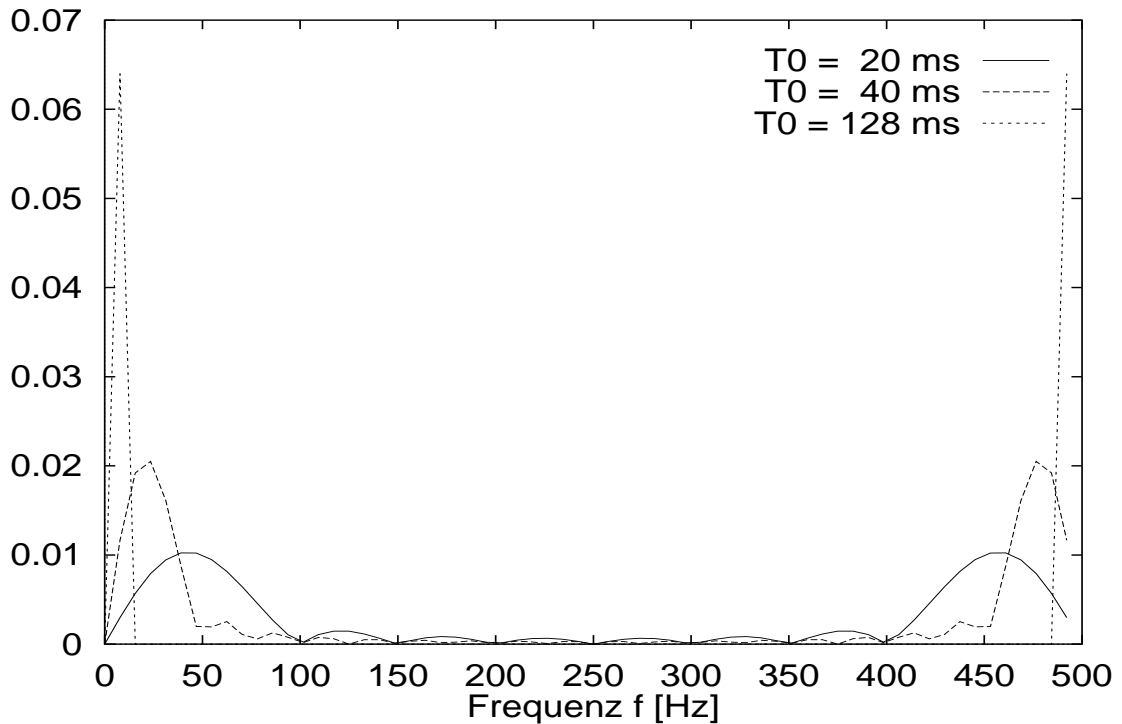
Wichtig: Der Algorithmus der FFT in *four1()* verlangt, daß die Anzahl der Daten eine Potenz von Zwei ist (2^n). Ist die Anzahl keine Potenz von Zwei, muß man das Feld bis zur nächsten Potenz mit Nullen auffüllen.

Normierung

Abweichend von der FORTRAN Funktion *FORK* muß man nach Anwendung von *four1()* die Daten multiplizieren mit:

Δt (Dem Samplingintervall) für aperiodische Signale
 $1.0 / \sqrt{\text{Wertezahl}}$ für periodische Signale

1.3 Plots für Aufgabe 1



Amplitudenspektren von verschiedenen Sinusfunktionen

1.4 Listen der FT's

Fourier-Transformierte des Sinussignals mit $T = 0.02s$

Frequenz	Realteil	Imaginärteil	Frequenz	Realteil	Imaginärteil
0	2.02656e-009	0	250.0	-2.02656e-009	0
7.8124995	0.00140319	0.00262519	257.8125	-0.000144791	-0.00027088
15.624999	0.0047315	0.00316148	265.625	-0.000454084	-0.000303407
23.437498	0.00787081	0.000775205	273.4375	-0.000659287	-6.49323e-005
31.249998	0.00873577	-0.00361847	281.25	-0.000579034	0.000239846
39.0625	0.00648952	-0.00790751	289.0625	-0.000279791	0.000340929
46.874996	0.00199275	-0.0100182	296.875	-2.72729e-005	0.000137114
54.687496	-0.00275116	-0.00906935	304.6875	-6.26158e-005	-0.000206418
62.499996	-0.00576768	-0.00576768	312.5	-0.00038769	-0.00038769
70.3125	-0.00616475	-0.00187006	320.3125	-0.000745811	-0.000226239
78.125	-0.0044618	0.000887506	328.12497	-0.000828695	0.000164838
85.937492	-0.00208061	0.00170751	335.93747	-0.000548614	0.000450238
93.749992	-0.000403812	0.000974886	343.74997	-0.000144462	0.000348766
101.56249	-2.17743e-005	-0.000221062	351.56247	-1.02725e-005	-0.000104316
109.37499	-0.000591033	-0.000884541	359.37497	-0.000361366	-0.000540825
117.18749	-0.001286	-0.000687377	367.18747	-0.00100806	-0.000538821
124.99999	-0.00145309	2.02656e-009	374.99997	-0.00145309	-2.02656e-009
132.8125	-0.00100806	0.000538821	382.81247	-0.001286	0.000687377
140.625	-0.000361366	0.000540825	390.62497	-0.000591033	0.000884541
148.4375	-1.02725e-005	0.000104316	398.43747	-2.17743e-005	0.000221062
156.25	-0.000144462	-0.000348766	406.24997	-0.000403812	-0.000974886
164.06248	-0.000548614	-0.000450238	414.06247	-0.00208061	-0.00170751
171.87498	-0.000828695	-0.000164838	421.87497	-0.0044618	-0.000887506
179.68748	-0.000745811	0.000226239	429.68747	-0.00616475	0.00187006
187.49998	-0.00038769	0.00038769	437.49997	-0.00576768	0.00576768
195.31248	-6.26158e-005	0.000206418	445.31247	-0.00275116	0.00906935
203.12498	-2.72729e-005	-0.000137114	453.12497	0.00199275	0.0100182
210.93748	-0.000279791	-0.000340929	460.93747	0.00648952	0.00790751
218.74998	-0.000579034	-0.000239846	468.74997	0.00873577	0.00361847
226.56248	-0.000659287	6.49323e-005	476.56247	0.00787081	-0.000775205
234.37498	-0.000454084	0.000303407	484.37497	0.0047315	-0.00316148
242.18748	-0.000144791	0.00027088	492.18747	0.00140319	-0.00262519

Fourier-Transformierte des Sinussignals mit $T = 0.04s$

Frequenz	Realteil	Imaginärteil	Frequenz	Realteil	Imaginärteil
0	2.02656e-009	0	250.0	2.02656e-009	0
7.8124995	0.00968253	0.00646966	257.8125	-0.000219536	-0.000146692
15.624999	0.0177446	-0.00735006	265.625	-0.000273068	0.000113107
23.437498	0.00399783	-0.0200985	273.4375	-1.23289e-005	6.19793e-005
31.249998	-0.0113705	-0.0113705	281.25	-0.000164815	-0.000164815
39.0625	-0.00859925	0.0017105	289.0625	-0.000324342	6.45151e-005
46.874996	-0.000756848	0.00182719	296.875	-5.07766e-005	0.000122582
54.687496	-0.00107142	-0.00160349	304.6875	-0.000110649	-0.000165594
62.499996	-0.00253345	2.02656e-009	312.5	-0.000372725	2.02656e-009
70.3125	-0.000602411	0.000901575	320.3125	-0.00012032	0.000180075
78.125	-0.000228852	-0.000552503	328.12497	-6.00715e-005	-0.000145029
85.937492	-0.00123944	-0.000246541	335.93747	-0.000417946	-8.31354e-005
93.749992	-0.000543688	0.000543688	343.74997	-0.000231693	0.000231693
101.56249	-3.55973e-005	-0.000178964	351.56247	-1.89486e-005	-9.52633e-005
109.37499	-0.000697812	-0.000289045	359.37497	-0.000460256	-0.000190646
117.18749	-0.000500883	0.000334677	367.18747	-0.000407285	0.000272137
124.99999	-2.02656e-009	0	374.99997	-2.02656e-009	0
132.8125	-0.000407285	-0.000272137	382.81247	-0.000500883	-0.000334677
140.625	-0.000460256	0.000190646	390.62497	-0.000697812	0.000289045
148.4375	-1.89486e-005	9.52633e-005	398.43747	-3.55973e-005	0.000178964
156.25	-0.000231693	-0.000231693	406.24997	-0.000543688	-0.000543688
164.06248	-0.000417946	8.31354e-005	414.06247	-0.00123944	0.000246541
171.87498	-6.00715e-005	0.000145029	421.87497	-0.000228852	0.000552503
179.68748	-0.00012032	-0.000180075	429.68747	-0.000602411	-0.000901575
187.49998	-0.000372725	-2.02656e-009	437.49997	-0.00253345	-2.02656e-009
195.31248	-0.000110649	0.000165594	445.31247	-0.00107142	0.00160349
203.12498	-5.07766e-005	-0.000122582	453.12497	-0.000756848	-0.00182719
210.93748	-0.000324342	-6.45151e-005	460.93747	-0.00859925	-0.0017105
218.74998	-0.000164815	0.000164815	468.74997	-0.0113705	0.0113705
226.56248	-1.23289e-005	-6.19793e-005	476.56247	0.00399783	0.0200985
234.37498	-0.000273068	-0.000113107	484.37497	0.0177446	0.00735006
242.18748	-0.000219536	0.000146692	492.18747	0.00968253	-0.00646966

Fourier-Transformierte des Sinus mit $T = 0.128s$

Frequenz	Realteil	Imaginärteil	Frequenz	Realteil	Imaginärteil
0	0	0	250.0	0	0
7.8124995	-6.16021e-012	-0.064	257.8125	-2.80297e-010	-9.53674e-009
15.624999	0	0	265.625	0	0
23.437498	-1.77518e-011	3.86309e-009	273.4375	-1.80872e-010	3.4996e-009
31.249998	0	0	281.25	0	0
39.0625	-6.25475e-011	1.94821e-009	289.0625	7.18713e-011	2.38278e-008
46.874996	0	0	296.875	0	0
54.687496	5.52312e-010	4.18606e-010	304.6875	-3.03102e-011	-2.75698e-009
62.499996	0	0	312.5	0	0
70.3125	-1.09793e-010	-5.6138e-010	320.3125	-2.98468e-010	-1.77699e-009
78.125	0	0	328.12497	0	0
85.937492	1.39698e-010	7.83276e-009	335.93747	7.01353e-011	-5.75243e-009
93.749992	0	0	343.74997	0	0
101.56249	-2.01884e-010	-2.64456e-009	351.56247	1.81351e-010	-2.24429e-009
109.37499	0	0	359.37497	0	0
117.18749	-1.06136e-009	6.69169e-009	367.18747	1.23408e-009	2.84505e-009
124.99999	0	0	374.99997	0	0
132.8125	1.23408e-009	-2.84505e-009	382.81247	-1.06136e-009	-6.69169e-009
140.625	0	0	390.62497	0	0
148.4375	1.81351e-010	2.24429e-009	398.43747	-2.01884e-010	2.64456e-009
156.25	0	0	406.24997	0	0
164.06248	7.01353e-011	5.75243e-009	414.06247	1.39698e-010	-7.83276e-009
171.87498	0	0	421.87497	0	0
179.68748	-2.98468e-010	1.77699e-009	429.68747	-1.09793e-010	5.6138e-010
187.49998	0	0	437.49997	0	0
195.31248	-3.03102e-011	2.75698e-009	445.31247	5.52312e-010	-4.18606e-010
203.12498	0	0	453.12497	0	0
210.93748	7.18714e-011	-2.38278e-008	460.93747	-6.25475e-011	-1.94821e-009
218.74998	0	0	468.74997	0	0
226.56248	-1.80872e-010	-3.4996e-009	476.56247	-1.77518e-011	-3.86309e-009
234.37498	0	0	484.37497	0	0
242.18748	-2.80297e-010	9.53674e-009	492.18747	-6.16008e-012	0.064

2 FFT von weiteren Impulsen

2.1 Aufgabenstellung

a) Wie Aufgabe1, Teil 1), für den Impuls

$$\begin{aligned} x(t) &= \exp\left(\left(-t/T_2\right)^2\right) \cdot \sin\left(\frac{2\pi}{T_0}\right) \cdot t && \text{für } 0 \leq t \leq T_1 \\ x(t) &= 0 && \text{sonst} \end{aligned}$$

mit $T_2 = LTT \cdot \Delta t$

Zahlenwerte:

$LX = 64$; $LT = 20$; $LTT = 10, 20, 40$; $\Delta t = 0.002$ s

b) Man bestimme wie auf S. 13 und 14 im Skript das Produkt von effektiver Impulsdauer Δt und der effektiven Breite Δf des Maximums der FT (Halbwertsbreite).

2.2 Listen der FT's

Fourier-Transformierte des Impulses mit $LTT = 10$

Frequenz	Realteil	Imaginärteil	Frequenz	Realteil	Imaginärteil
0	0.0080719	0	249.99998	-0.000156408	0
7.8124995	0.00815496	-0.00260527	257.8125	-0.00016073	-2.56705e-006
15.624999	0.00682579	-0.00628691	265.625	-0.000162726	2.07591e-006
23.437498	0.00264574	-0.00876004	273.4375	-0.000160153	9.29356e-007
31.249998	-0.00209125	-0.00758376	281.25	-0.000166014	-2.89893e-006
39.0625	-0.00417669	-0.00408604	289.0625	-0.000172251	1.41335e-006
46.874996	-0.00356729	-0.00132182	296.875	-0.000172111	1.87433e-006
54.687496	-0.002324	-0.000271543	304.6875	-0.000179781	-3.05736e-006
62.499996	-0.00154796	-4.91659e-005	312.5	-0.000191085	5.57805e-007
70.3125	-0.00111183	9.29509e-006	320.3125	-0.00019475	2.84365e-006
78.125	-0.000839515	3.47637e-007	328.12497	-0.000205235	-3.03688e-006
85.937492	-0.00069129	-9.18701e-006	335.93747	-0.00022345	-5.46428e-007
93.749992	-0.000576824	7.05473e-006	343.74997	-0.000233557	3.88177e-006
101.56249	-0.00047913	1.42538e-006	351.56247	-0.000249291	-2.79152e-006
109.37499	-0.000424184	-6.61872e-006	359.37497	-0.000278399	-1.9994e-006
117.18749	-0.000376907	4.01163e-006	367.18747	-0.000300445	5.06783e-006
124.99999	-0.000327392	2.26784e-006	374.99997	-0.000327392	-2.26784e-006
132.8125	-0.000300445	-5.06783e-006	382.81247	-0.000376907	-4.01163e-006
140.625	-0.000278399	1.9994e-006	390.62497	-0.000424184	6.61872e-006
148.4375	-0.000249291	2.79152e-006	398.43747	-0.00047913	-1.42538e-006
156.25	-0.000233557	-3.88177e-006	406.24997	-0.000576824	-7.05473e-006
164.06248	-0.00022345	5.46428e-007	414.06247	-0.00069129	9.18701e-006
171.87498	-0.000205235	3.03688e-006	421.87497	-0.000839515	-3.47637e-007
179.68748	-0.00019475	-2.84365e-006	429.68747	-0.00111183	-9.29509e-006
187.49998	-0.000191085	-5.57805e-007	437.49997	-0.00154796	4.91659e-005
195.31248	-0.000179781	3.05736e-006	445.31247	-0.002324	0.000271543
203.12498	-0.000172111	-1.87433e-006	453.12497	-0.00356729	0.00132182
210.93748	-0.000172251	-1.41335e-006	460.93747	-0.00417669	0.00408604
218.74998	-0.000166014	2.89893e-006	468.74997	-0.00209125	0.00758376
226.56248	-0.000160153	-9.29356e-007	476.56247	0.00264574	0.00876004
234.37498	-0.000162726	-2.07591e-006	484.37497	0.00682579	0.00628691
242.18748	-0.00016073	2.56705e-006	492.18747	0.00815496	0.00260527

Fourier-Transformierte des Impulses mit $LTT = 20$

Frequenz	Realteil	Imaginärteil	Frequenz	Realteil	Imaginärteil
0	0.00454074	0	249.99998	-0.000100392	0
7.8124995	0.00927063	0.00168566	257.8125	-0.000181575	-5.3784e-005
15.624999	0.0125522	-0.00708143	265.625	-0.000201305	4.19738e-005
23.437498	0.00357524	-0.0149598	273.4375	-0.000106864	2.19297e-005
31.249998	-0.0070432	-0.0102477	281.25	-0.000166291	-6.05445e-005
39.0625	-0.00729037	-0.00134496	289.0625	-0.000225444	2.51719e-005
46.874996	-0.00264048	0.000902087	296.875	-0.000127217	4.36107e-005
54.687496	-0.00139332	-0.00046165	304.6875	-0.000156619	-6.17391e-005
62.499996	-0.001762	-0.000138521	312.5	-0.000255151	2.92866e-006
70.3125	-0.000950164	0.000356859	320.3125	-0.00016472	6.47543e-005
78.125	-0.000553904	-0.000155899	328.12497	-0.00015534	-5.6177e-005
85.937492	-0.000852614	-0.000130026	335.93747	-0.000293377	-2.59098e-005
93.749992	-0.00058291	0.000197615	343.74997	-0.000226499	8.49648e-005
101.56249	-0.000305125	-4.20079e-005	351.56247	-0.000168239	-4.14634e-005
109.37499	-0.000503601	-0.000119367	359.37497	-0.000346593	-6.38188e-005
117.18749	-0.000429748	0.000116318	367.18747	-0.000327702	0.000103583
124.99999	-0.00020788	1.26281e-005	374.99997	-0.00020788	-1.26281e-005
132.8125	-0.000327702	-0.000103583	382.81247	-0.000429748	-0.000116318
140.625	-0.000346593	6.38188e-005	390.62497	-0.000503601	0.000119367
148.4375	-0.000168239	4.14634e-005	398.43747	-0.000305125	4.20079e-005
156.25	-0.000226499	-8.49648e-005	406.24997	-0.00058291	-0.000197615
164.06248	-0.000293377	2.59098e-005	414.06247	-0.00085291	0.000130026
171.87498	-0.00015534	5.6177e-005	421.87497	-0.000553904	0.000155899
179.68748	-0.00016472	-6.47543e-005	429.68747	-0.000950164	-0.000356859
187.49998	-0.000255151	-2.92866e-006	437.49997	-0.001762	0.000138521
195.31248	-0.000156619	6.17391e-005	445.31247	-0.00139332	0.00046165
203.12498	-0.000127217	-4.36107e-005	453.12497	-0.00264048	-0.000902087
210.93748	-0.000225444	-2.51719e-005	460.93747	-0.00729037	0.00134496
218.74998	-0.000166291	6.05445e-005	468.74997	-0.0070432	0.0102477
226.56248	-0.000106864	-2.19297e-005	476.56247	0.00357524	0.0149598
234.37498	-0.000201305	-4.19738e-005	484.37497	0.0125522	0.00708143
242.18748	-0.000181575	5.3784e-005	492.18747	0.00927063	-0.00168566

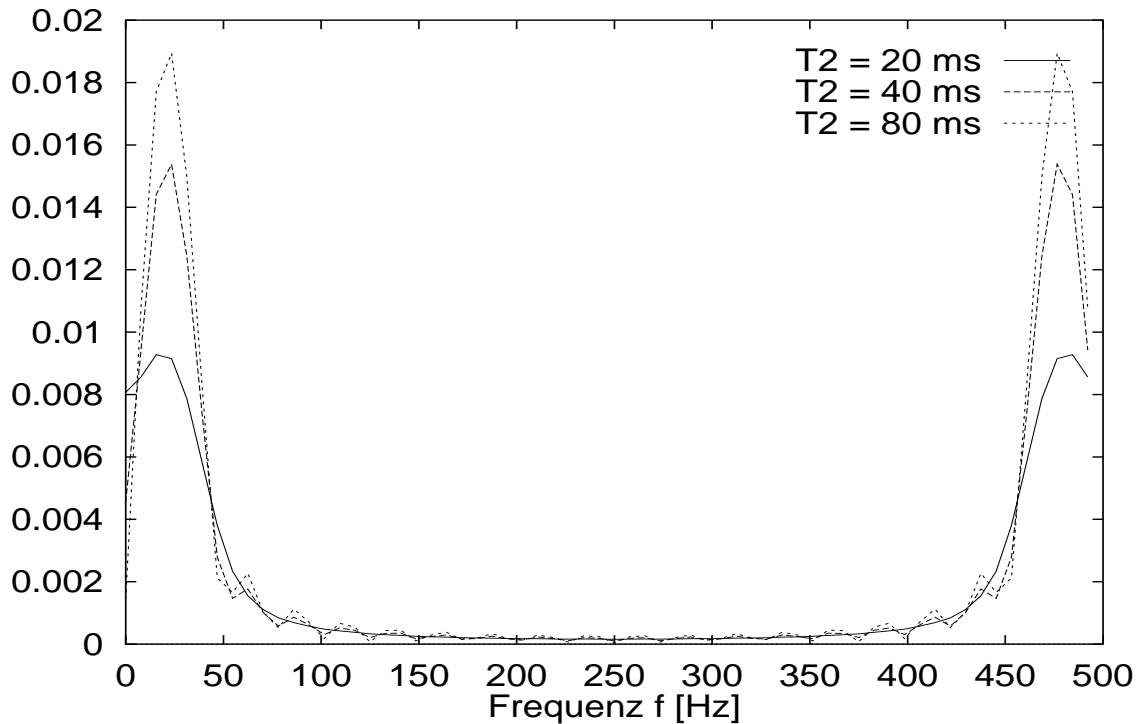
Fourier-Transformierte des Impulses mit $LTT = 40$

Frequenz	Realteil	Imaginärteil	Frequenz	Realteil	Imaginärteil
0	0.00144895	0	249.99998	-3.50601e-005	0
7.8124995	0.00957192	0.00496269	257.8125	-0.000206298	-0.000114205
15.624999	0.0161459	-0.00727187	265.625	-0.000247894	8.83234e-005
23.437498	0.00390193	-0.0185349	273.4375	-4.52793e-005	4.78344e-005
31.249998	-0.0100446	-0.0110974	281.25	-0.00016553	-0.000128382
39.0625	-0.00828123	0.000746993	289.0625	-0.000289656	5.10343e-005
46.874996	-0.00137988	0.00161252	296.875	-7.72001e-005	9.47443e-005
54.687496	-0.00112423	-0.00120974	304.6875	-0.000127075	-0.000129484
62.499996	-0.00226876	-7.27102e-005	312.5	-0.000331626	1.55021e-006
70.3125	-0.000733604	0.000722287	320.3125	-0.000135197	0.00013957
78.125	-0.000330912	-0.000408473	328.12497	-9.38044e-005	-0.000114547
85.937492	-0.00110208	-0.000213813	335.93747	-0.00037476	-6.23438e-005
93.749992	-0.000562568	0.000424033	343.74997	-0.000228745	0.00018051
101.56249	-0.000126809	-0.000127304	351.56247	-7.13995e-005	-7.7684e-005
109.37499	-0.000627567	-0.00023266	359.37497	-0.000421624	-0.000145367
117.18749	-0.00047853	0.000257701	367.18747	-0.000377713	0.000214153
124.99999	-7.21503e-00	5.667393e-006	374.99997	-7.21503e-005	-6.67393e-006
132.8125	-0.000377713	-0.000214153	382.81247	-0.00047853	-0.000257701
140.625	-0.000421624	0.000145367	390.62497	-0.000627567	0.00023266
148.4375	-7.13995e-00	5.7684e-005	398.43747	-0.000126809	0.000127304
156.25	-0.000228745	-0.00018051	406.24997	-0.000562568	-0.000424033
164.06248	-0.00037476	6.23438e-005	414.06247	-0.00110208	0.000213813
171.87498	-9.38044e-00	5.0000114547	421.87497	-0.000330912	0.000408473
179.68748	-0.000135197	-0.00013957	429.68747	-0.000733604	-0.000722287
187.49998	-0.000331626	-1.55021e-006	437.49997	-0.00226876	7.27102e-005
195.31248	-0.000127075	0.000129484	445.31247	-0.00112423	0.00120974
203.12498	-7.72001e-00	5.947443e-005	453.12497	-0.00137988	-0.00161252
210.93748	-0.000289656	-5.10343e-005	460.93747	-0.00828123	-0.000746993
218.74998	-0.00016553	0.000128382	468.74997	-0.0100446	0.0110974
226.56248	-4.52793e-00	5.478344e-005	476.56247	0.00390193	0.0185349
234.37498	-0.000247894	-8.83234e-005	484.37497	0.0161459	0.0072187
242.18748	-0.000206298	0.000114205	492.18747	0.00957192	-0.00496269

2.3 Bestimmung von $\Delta T \cdot \Delta f$

...

2.4 Plots für Aufgabe 2



3 Synthetisches Seismogramm

3.1 Aufgabenstellung

1) Man schreibe ein FORTRAN / C Programm für synthetische Seismogramme, basierend auf Gleichung (2-47)¹ und dem Matrizenformalismus Gl. (2-43), (2-44) und (2-45) für ein beliebig geschichtetes Medium. Das Programm soll erzeugen:

1a) 2 Plots der Amplitudenspektren von Eingangssignal $F(t)$ und synthetischem Seismogramm $u_r(0, t)$ sowie des Betrags der Reflektivität $R(\omega)$, bis zur Nyquistfrequenz.

1b) Plot des synthetischen Seismogramms

Die Multiplikation $\bar{F}(\omega) \cdot R(\omega)$ wird **nur bis zur Nyquistfrequenz** durchgeführt; bis zur Abtastfrequenz wird ergänzt wie auf S. 26 im Skript beschrieben.

¹Hinweise auf nummerierte Gleichungen beziehen sich immer auf das Skript „Digitale Signalverarbeitung“ von Gerhard Müller (Frankfurt, 1994).

2) Anwendung:

2a) Signal $F(t)$ wie in Aufgabe 1, Teil 1), mit

$$\Delta t = 0.002s, \quad LX = 128, \quad LT = 20$$

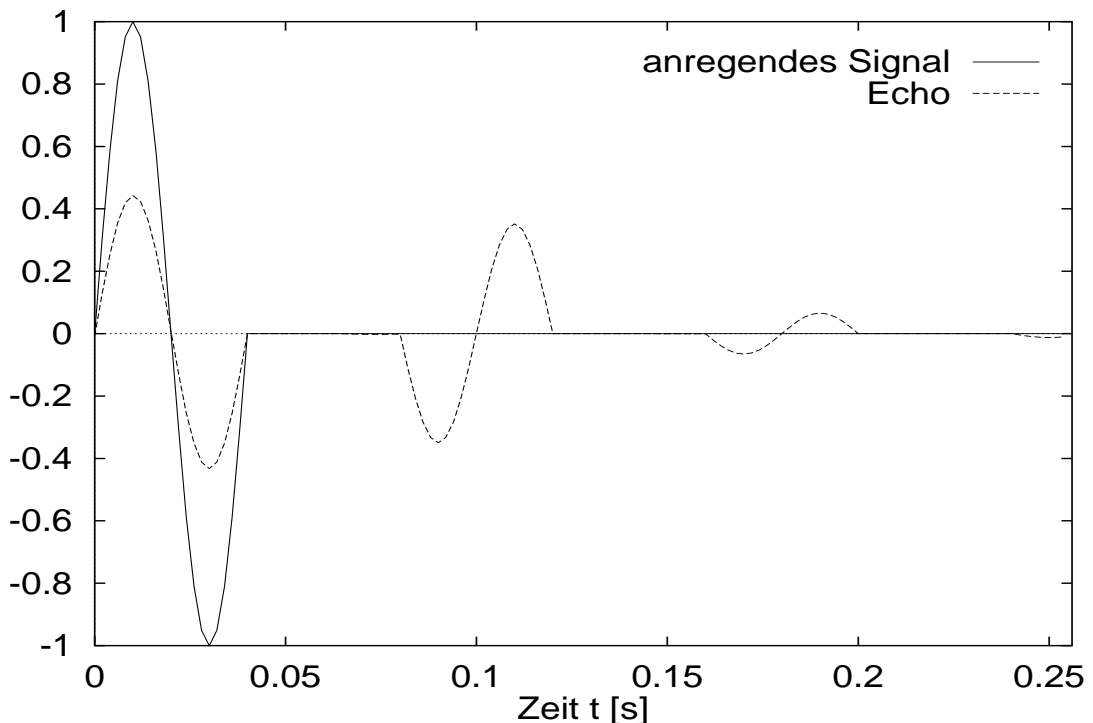
2b) Schichtenmodell:

Index l	d_l [m]	α_l [m/s]	ρ_l [g/cm ³]
1	100	4800	2.5
2	100	2500	1.9
3	bel.	4800	2.5

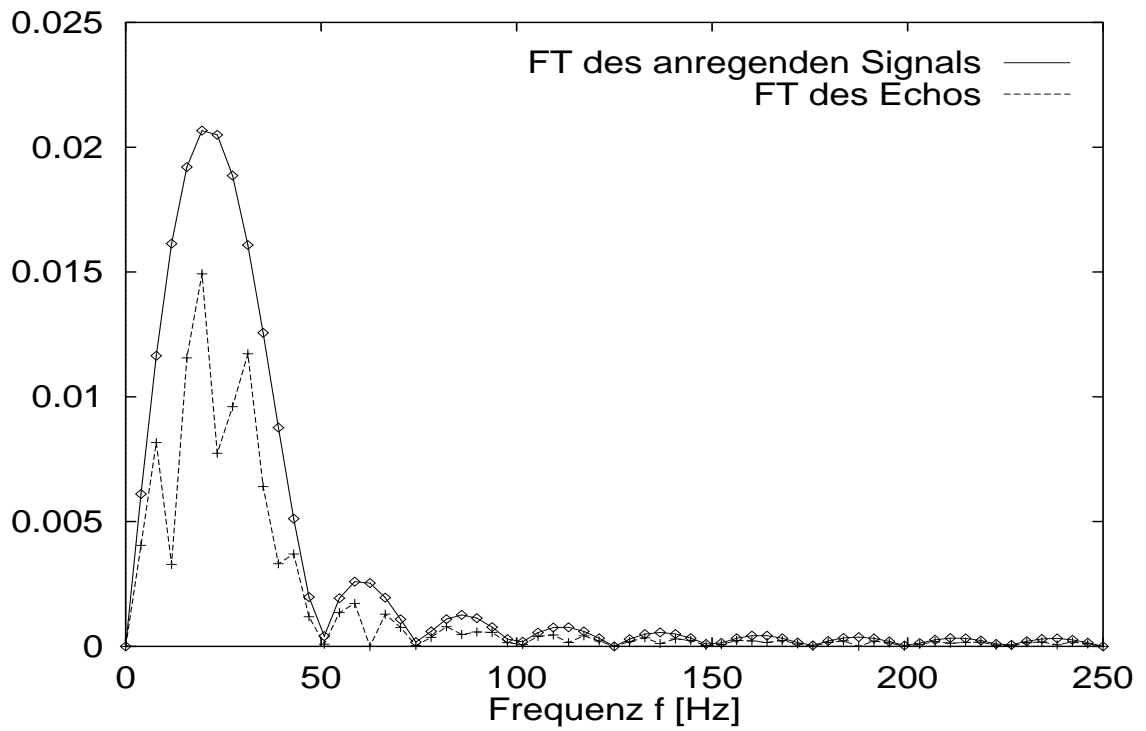
2c) Rechnung für Signal $F(t) = \delta(t)$ mit der Zeitreihe $(0,0,1/\Delta t,0,0,\dots)$ (Die führenden Nullen sollen verhindern, daß beim Plotten der Impuls durch die X-Achse nicht verdeckt wird.)

Man überlege, was man bei dem Modell unter b) erwarten muß (Primärreflexionen, Multiple, Laufzeiten, Polaritäten, Amplituden, vgl. (2-48) im Skript).

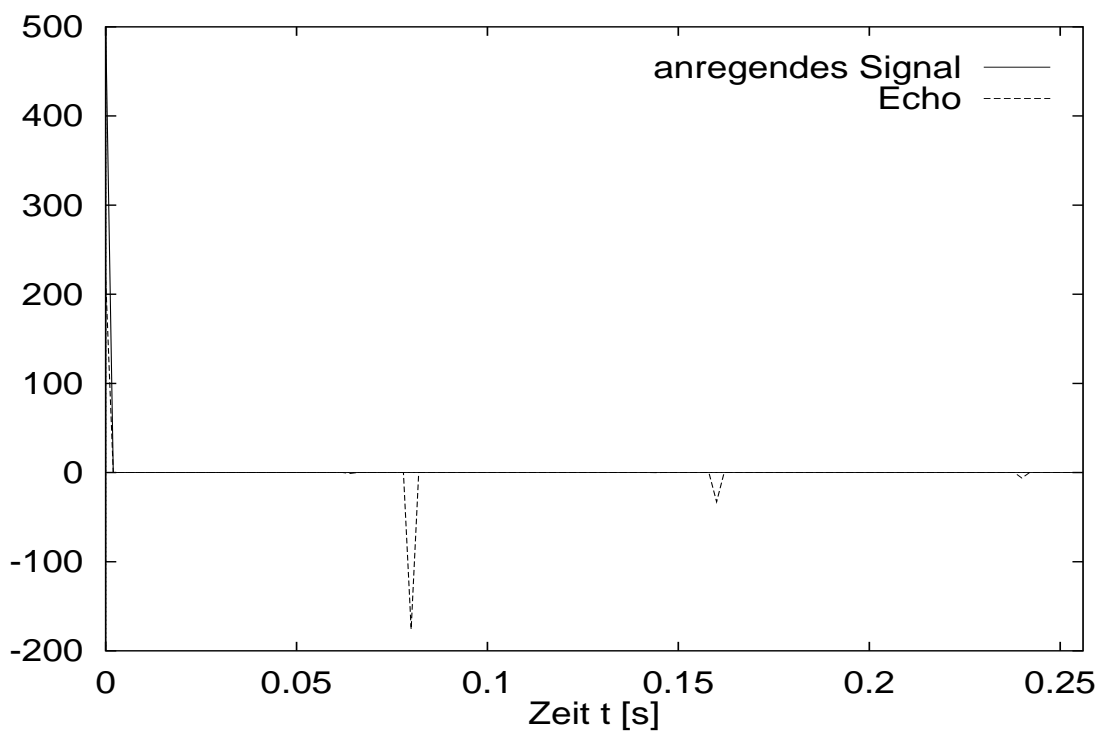
3.2 Plots für Aufgabe 3



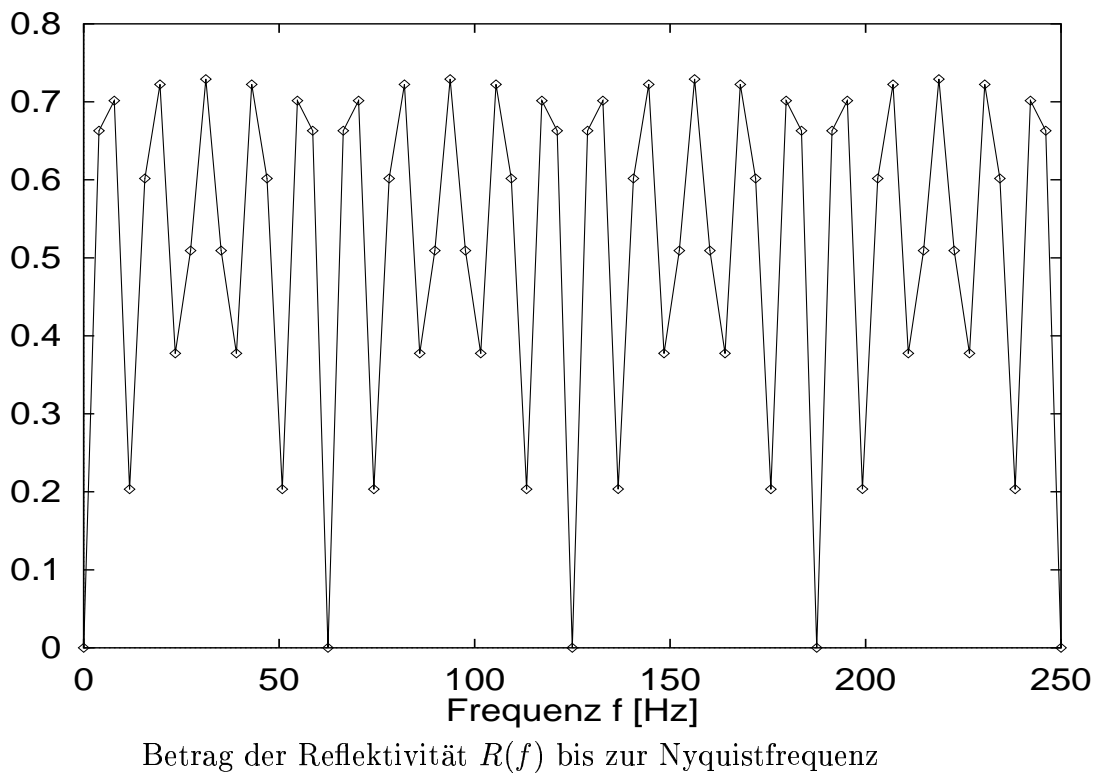
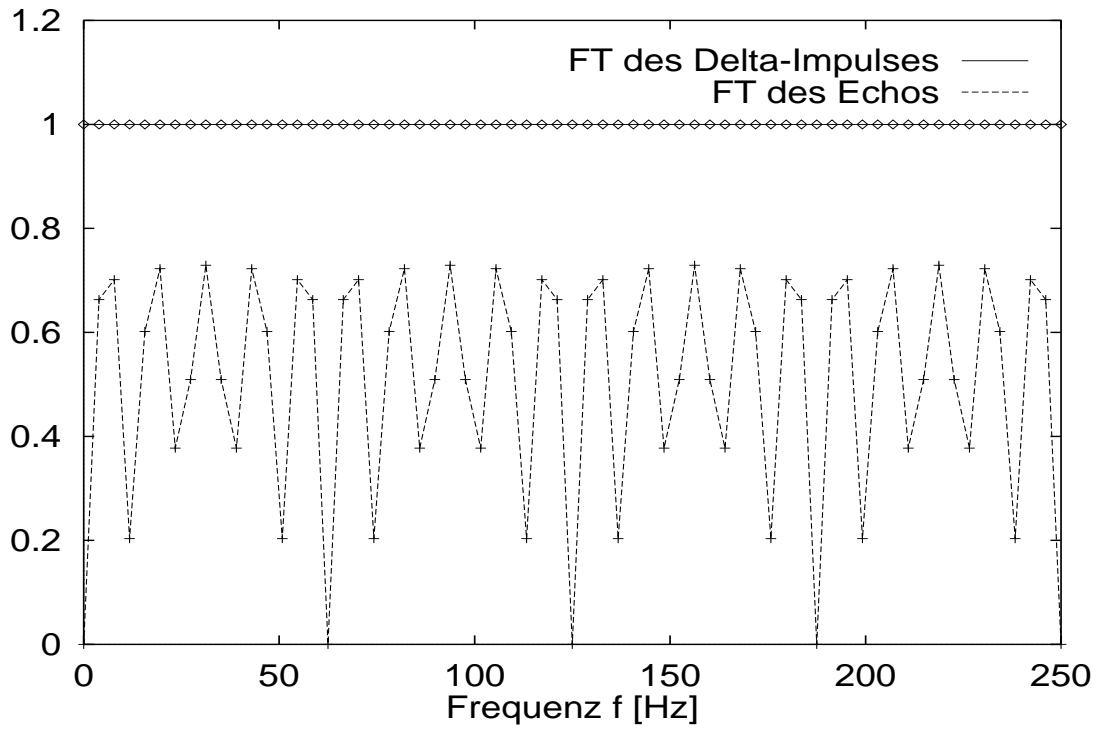
Sinus-Signal und dessen Seismogramm für zwei Grenzschichten
(Seismogramm um Laufzeit des ersten Echos zeitverschoben)



Betrag der FT des anregenden Sinussignals und des Seismogramms



Delta-Impuls und dessen Seismogramm für zwei Grenzschichten



3.3 Überlegungen zum Seismogramm

Das erste Echo stammt von der Reflexion der ersten Grenzschicht. Dort ist ein Übergang zu einem langsameren Medium, die *Polarität* der Welle bleibt also erhalten. Die *Laufzeit* beträgt $t_1 = 2 \cdot 100/4800 \text{ s} = 0.0417 \text{ s}$.

Die nächste Reflexion stammt von der zweiten Grenzschicht, einem Übergang zu einem schnelleren Medium, die *Polarität* kehrt sich um. Die *Laufzeit* beträgt $t_2 = t_1 + 2 \cdot 100/2500 \text{ s} = 0.0417 \text{ s} + 0.08 \text{ s} = 0.1217 \text{ s}$.

Alle weiteren Reflexionen sind *Multiple* der zweiten Reflexion, die mehrfach zwischen 1. und 2. Grenzschicht reflektiert werden. Da immer $(2 \cdot n) + 1$ Reflexionen an Grenzschichten zu schnelleren Medien auftreten, sind die *Polaritäten* aller Multiplen umgekehrt. Die Laufzeiten der Multiplen betragen $t_n = t_1 + 2n \cdot 100/2500 \text{ s} = 0.0417 \text{ s} + n \cdot 0.08 \text{ s}$

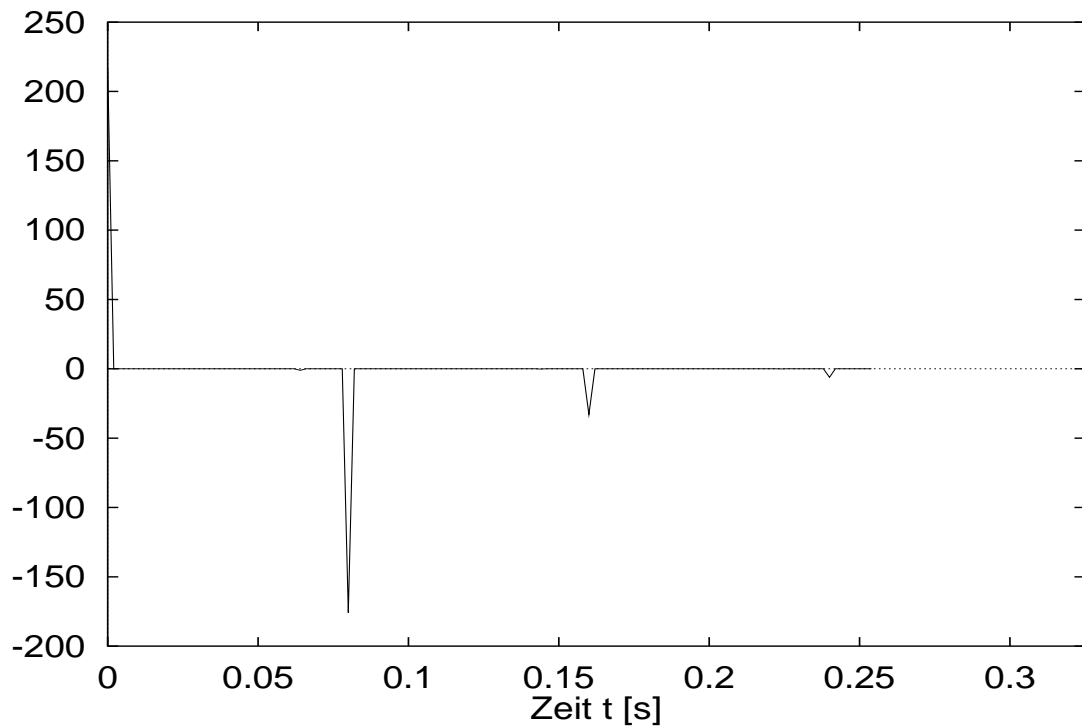
Amplituden ...

4 Faltung

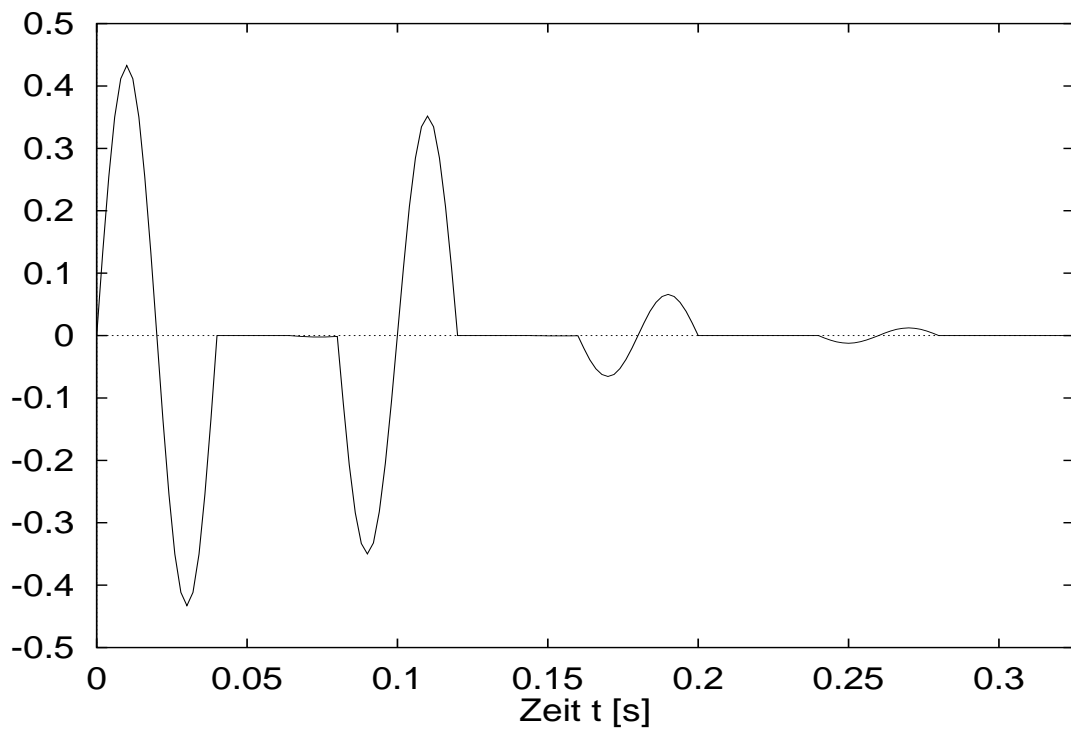
4.1 Aufgabenstellung

Man schreibe ein Unterprogramm für die Faltung, basierend auf Formel (3-3) im Skript. Dann wiederhole man Aufgabe 3, Teil 2c) (ergibt das Impulsseismogramm), und falte das Resultat mit dem Eingangssignal von Teil 2a). Es muß sich dasselbe Seismogramm wie in Aufgabe 3 ergeben.

4.2 Plots für Aufgabe 4



Impulse seismogramm zeitlich verschoben um die Laufzeit des ersten Echos



Faltung von Sinus-Signal mit Impulse seismogramm

5 Aliasing

5.1 Aufgabenstellung

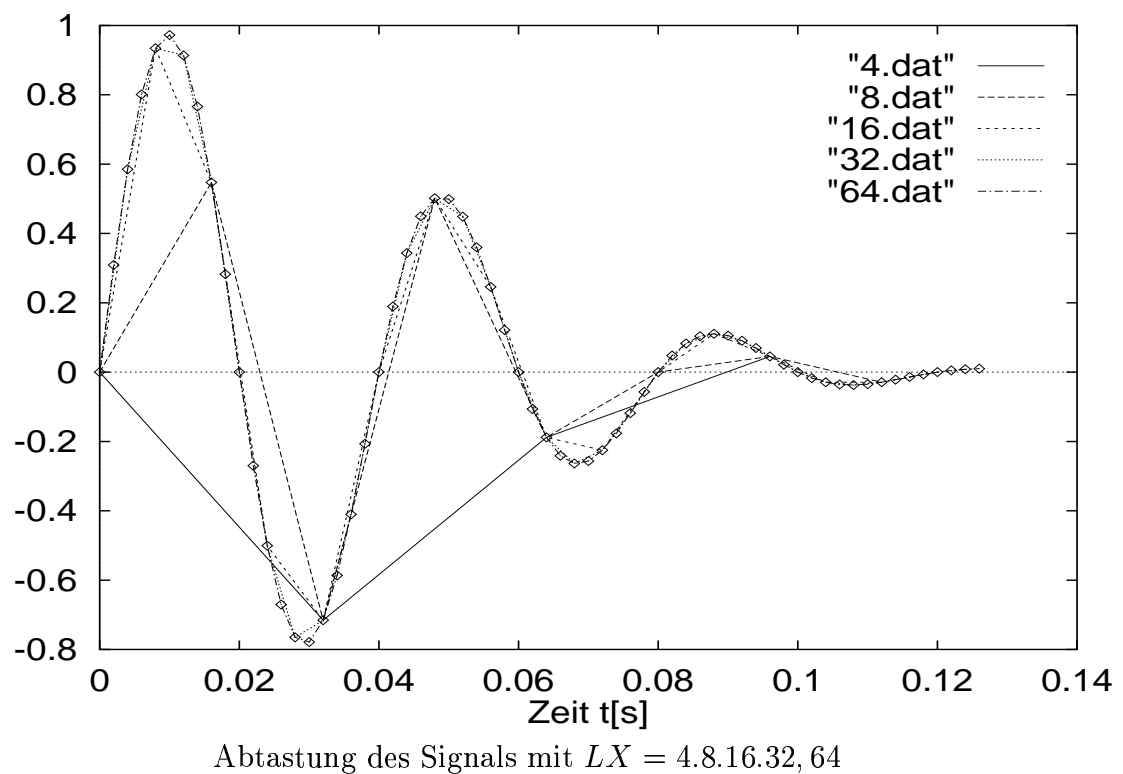
Man erzeuge und diskutiere Aliasing, indem man das Spektrum des Impulses

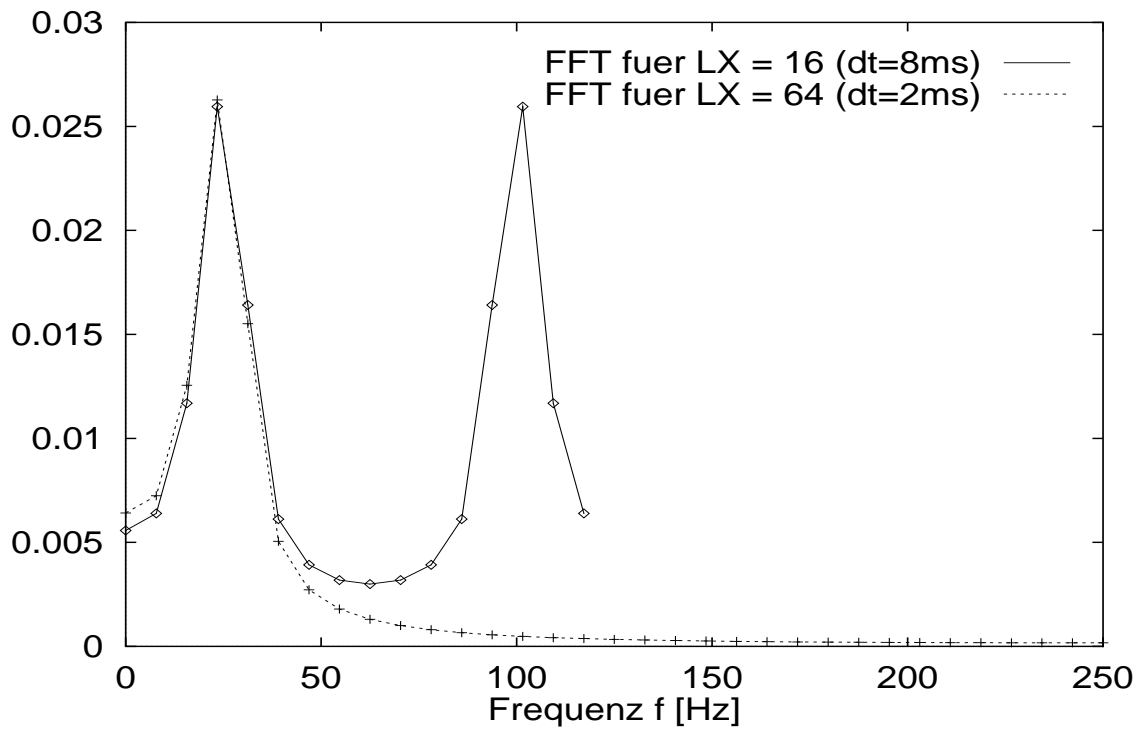
$$x(t) = \sin\left(\frac{2\pi}{T_0} \cdot t\right) \cdot \exp\left(\frac{-t^2}{T_1^2}\right)$$

$$\text{mit } \begin{array}{ll} T_0 & = 0.04 \text{ s} \\ LX \cdot \Delta t & = 0.128 \text{ s} \end{array} \quad \begin{array}{ll} T_1 & = 0.06 \text{ s} \\ LX & = 64, 32, \dots, 4 \end{array}$$

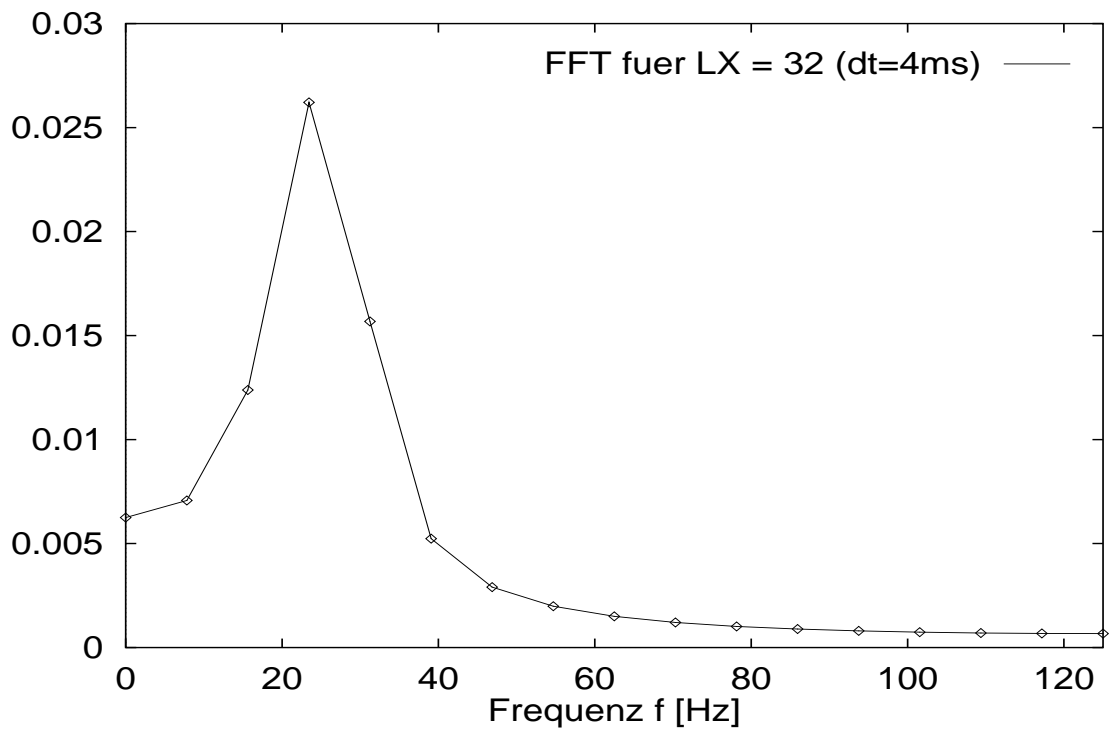
Man plote in jedem Fall $x(t)$ und den Betrag des Spektrums, letzteren bis zur Abtastfrequenz. Man übertrage das exakte Spektrum ($LX = 64$) in den Plot für $LX = 16$ und vergleiche.

5.2 Plots für Aufgabe 5

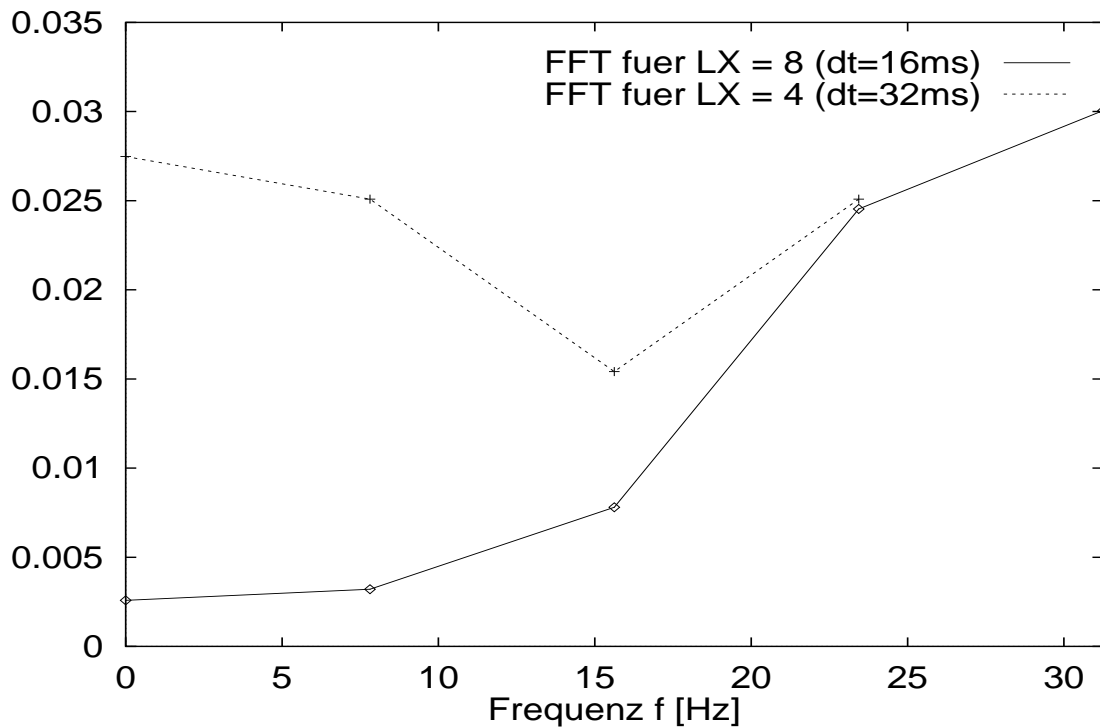




Betrag der FT des Signals, exakt und mit Aliasing
 Nyquistfrequenz: 62.5 Hz für LX=16 und 250 Hz für LX=64



Betrag der FT mit Aliasing (Nyquistfreq 125 Hz)



Betrag der FT mit Aliasing

Nyquistfrequenz: 31.25 Hz für LX=8 und 15.625 Hz für LX=4

6 Tiefpaßfilterung

6.1 Aufgabenstellung:

- 1) Man schreibe ein FORTRAN / C - Programm, das den halben Filteroperator eines Tiefpaßfilters berechnet und plottet zu vorgegebenen Werten von ω_N (oder f_N bzw. Δt), ω_L (oder f_L) und κ , und wahlweise für das Rechteck- oder Cosinus-Fenster.

Der Betrag der Übertragungsfunktion soll ebenfalls geplottet werden. Vor Anwendung von *FORK* / *four1()* Verschiebung des Filteroperators nach rechts um $\kappa \cdot \Delta t$.

- 2) Numerische Rechnungen für folgende Zahlenwerte:

$$\Delta t = 0.002s, \quad f_L = 100Hz, \quad \kappa = 10, 20, 30$$

Man diskutiere qualitativ den Einfluß des Fensters und von κ auf die Übertragungsfunktion. Was ist der Hauptnachteil des Rechteck-Fensters?

- 3) Für $\kappa = 10$ und das Cosinus-Fenster zeichne man die Übertragungsfunktion doppeltlogarithmisch und bestimme den 3db-Punkt (d.h. seine Fre-

quenz) und die Flankensteilheit in db/Oktave zwischen 100 und 110 Hz.

4) Tiefpaßfilterung von synthetischen Seismogrammen:

4a) Synthetisches Seismogramm wie in Aufgabe 3, Teil 2), berechnen, hier aber mit

$$\Delta t = 0.001 \text{ s}, \quad LX = 256, \quad LT = 40$$

4b) Überlagerung einer Störschwingung mit Frequenz 130 Hz und Amplitude 0.5, Plot des gestörten Seismogramms

4c) Tiefpaß-Unterprogramm einbauen und Filteroperator berechnen für

$$\Delta t = 0.001 \text{ s}, \quad f_l = 100 \text{ Hz}, \quad \kappa = 201,$$

und Cosinus-Fenster, Plot von Filteroperator und Übertragungsfunktion

4d) Faltung des gestörten Seismogramms mit dem Filteroperator, dabei Verwendung des Faltungsprogramms aus Aufgabe 4, Plot des gefilterten Seismogramms. Warum sind die Reflexionen zeitlich verschoben?

6.2 Erläuterungen

Zu Teil 2) Was ist der Hauptnachteil des Rechteck-Fensters?

Der Filter im Zeitbereich ist eine *sinc*-Funktion, die langsam abklingt. Das Rechteck-Fenster schneidet diese plötzlich ab, obwohl die folgenden Filterkoeffizienten noch Beiträge liefern. Die Folge ist eine starke Welligkeit der Übertragungsfunktion des Filters im Durchlaßbereich

Zu Teil 3) - 3db-Punkt Der 3db-Punkt ist ausgezeichnet durch eine Abnahme der Amplitude um einen Faktor von 0.7079. Die berechnete Übertragungsfunktion hat um den 3db-Punkt die Werte:

$f[\text{Hz}]$	$F(f)$
85.937492	0.762993
93.749992	0.623384

Der 3db-Punkt liegt damit bei $f = 89.0$ Hz. (linear Interpoliert)

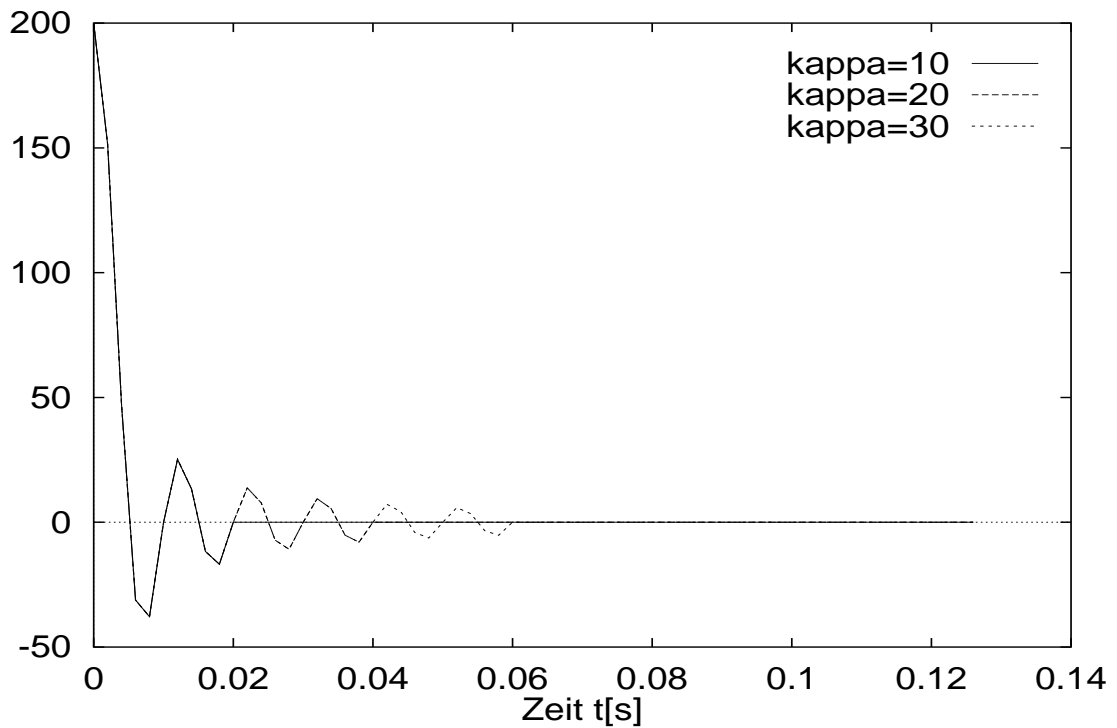
Die Flankensteilheit im Bereich zw. 100 und 110 Hz erhält man aus der Steigung der doppeltlogarithmischen Darstellung des Filters durch Multiplikation mit 6.02. (siehe Skript S. 52)

$f[\text{Hz}]$	$F(f)$	$\ln(f)$	$\ln(F(f))$
101.56249	0.468845	4.621	-0.757
109.37499	0.318079	4.695	-1.145

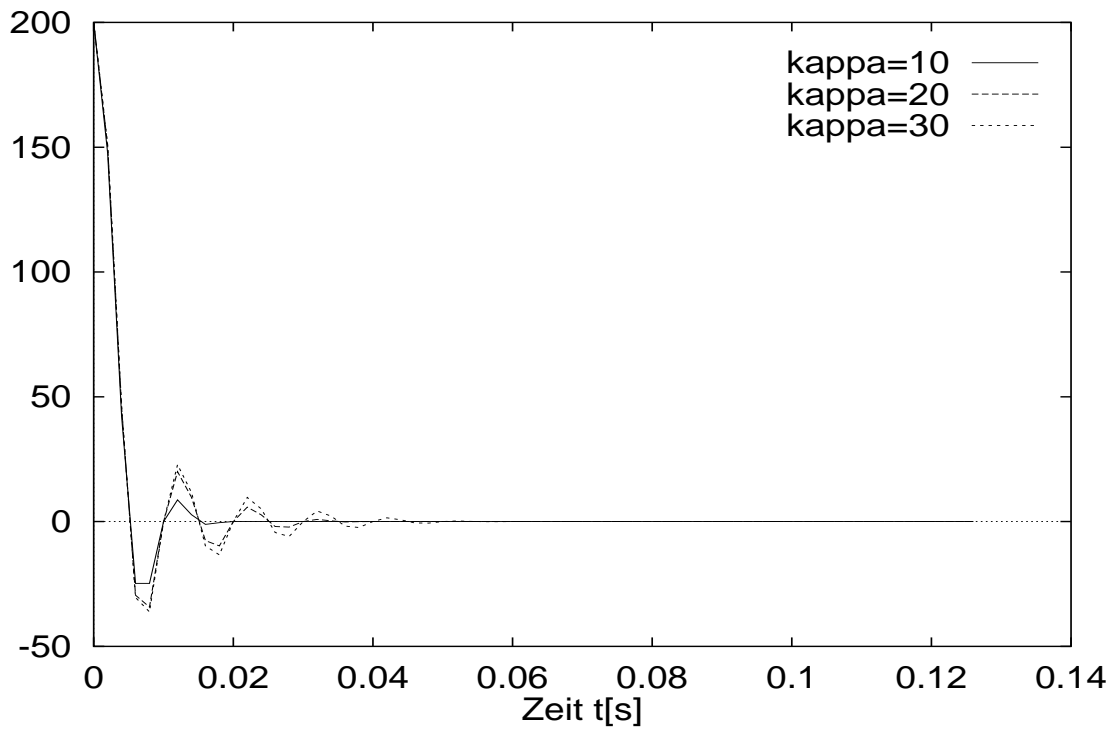
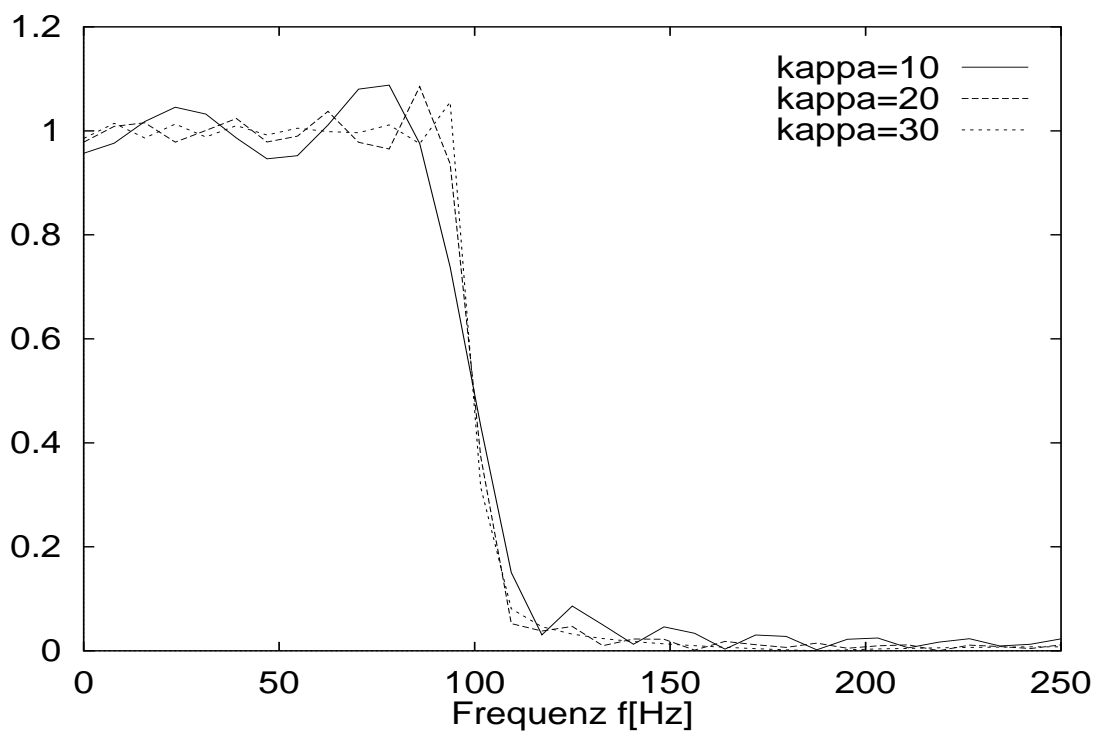
Damit beträgt die Flankensteilheit zwischen 100 und 110 Hz
0.798 db/Oktave. (???)

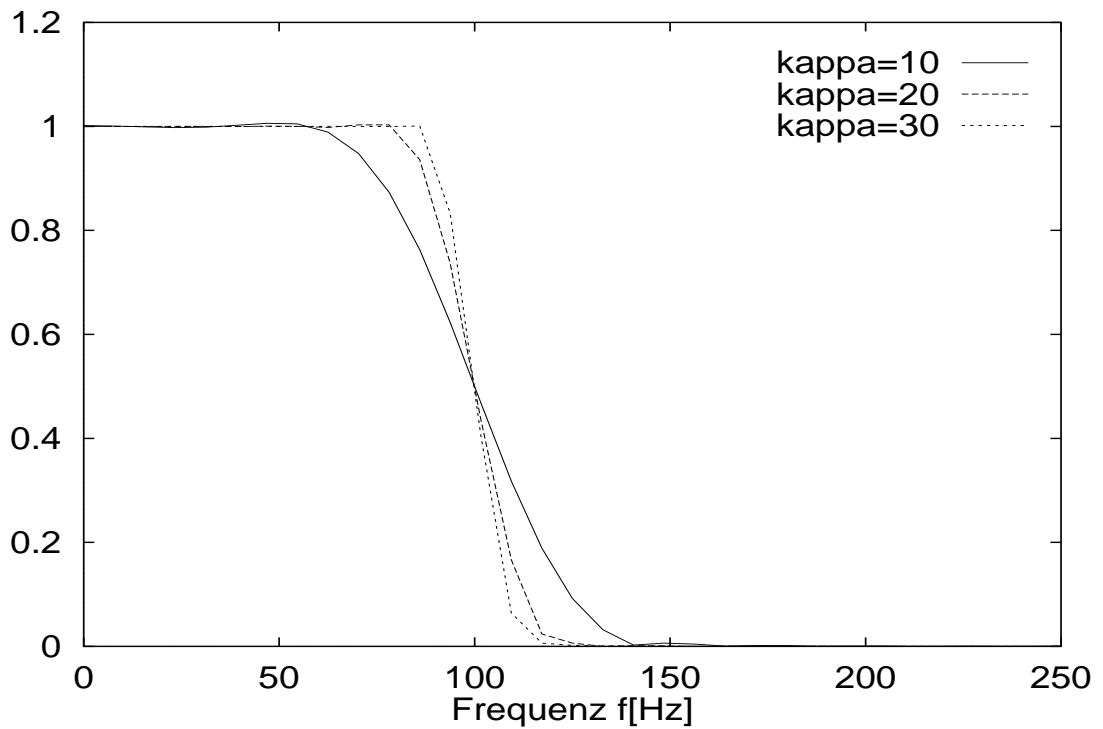
ZuTeil 4d) Das Maximum des Filters im Zeitbereich liegt nicht am Anfang, sondern in der Mitte des Wavelets. Durch die Faltung des Seismogramms mit dem Filter verschiebt sich das gefilterte Seismogramm genau um die Zeit, bei der das Maximum des Filters liegt.

6.3 Plots für Aufgabe 6:

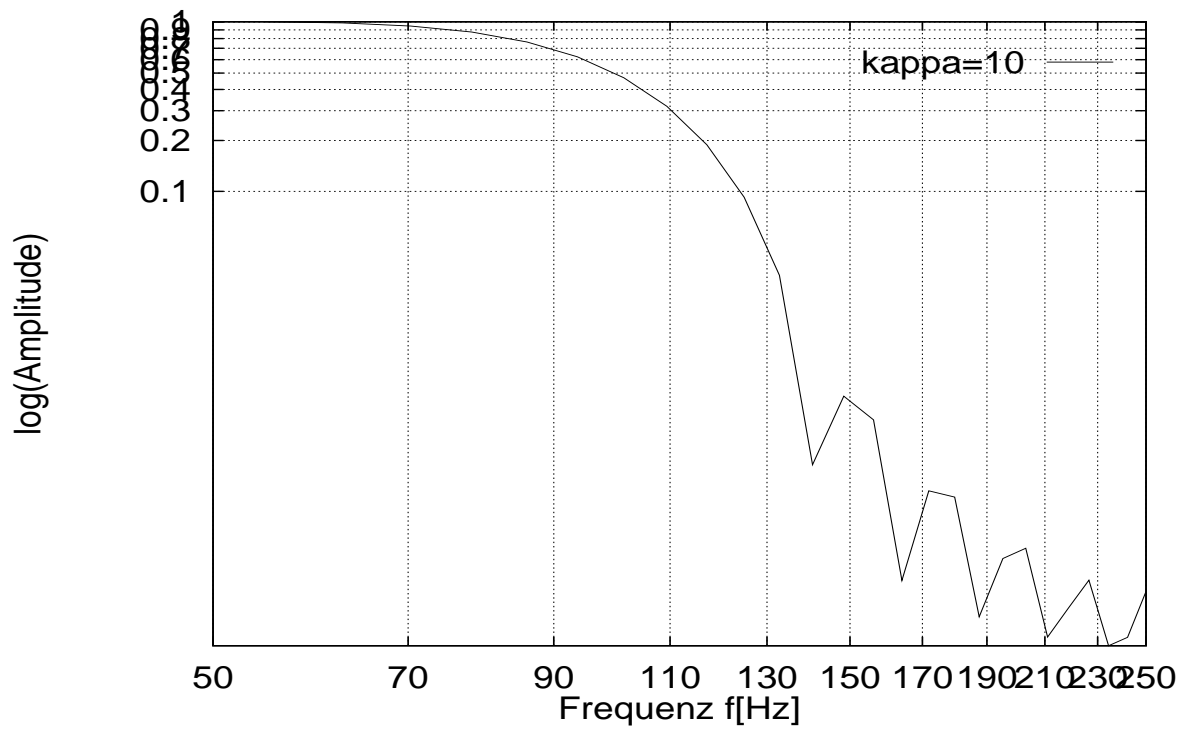


2a) Halber Filteroperator für $dt = 2 \text{ ms}$, $f_L = 100 \text{ Hz}$ und Rechteckfenster

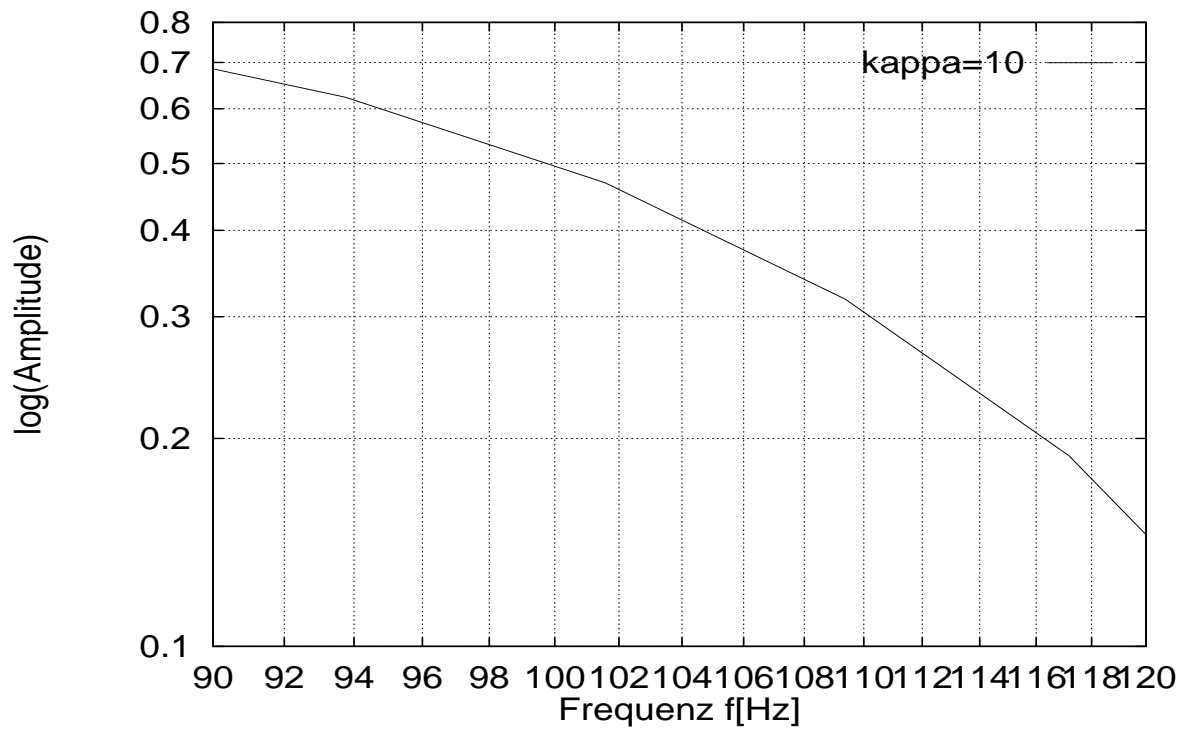
Halber Filteroperator für $dt = 2$ ms, $f_L = 100$ Hz und Cosinusfenster2b) Betrag der Übertragungsfunktion für $dt = 2$ ms, $f_L = 100$ Hz und Rechteckfenster



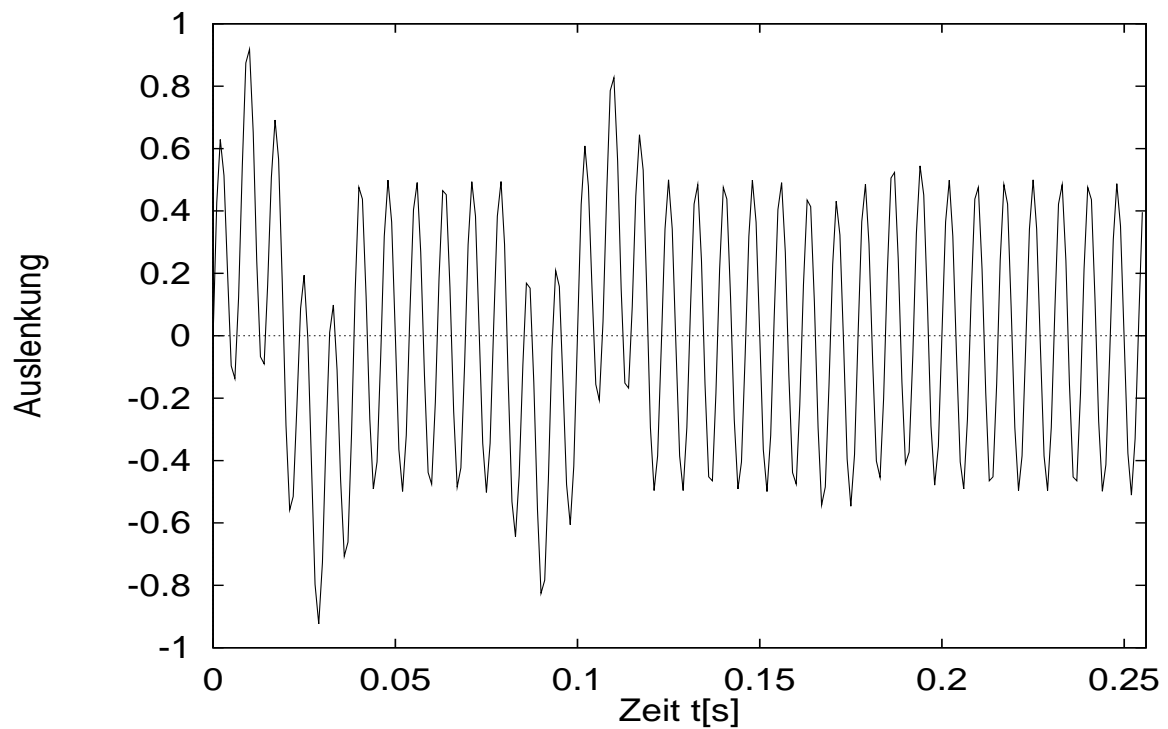
2b) Betrag der Übertragungsfunktion für $dt = 2$ ms, $f_L = 100$ Hz und Cosinusfenster



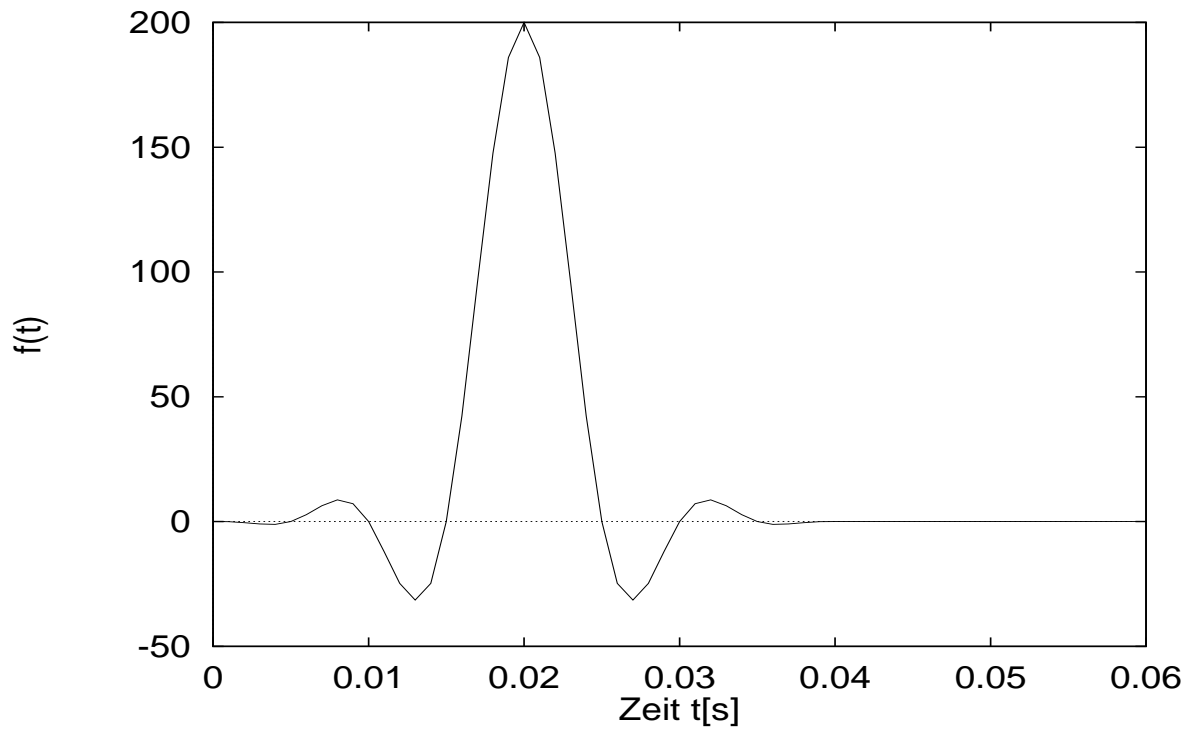
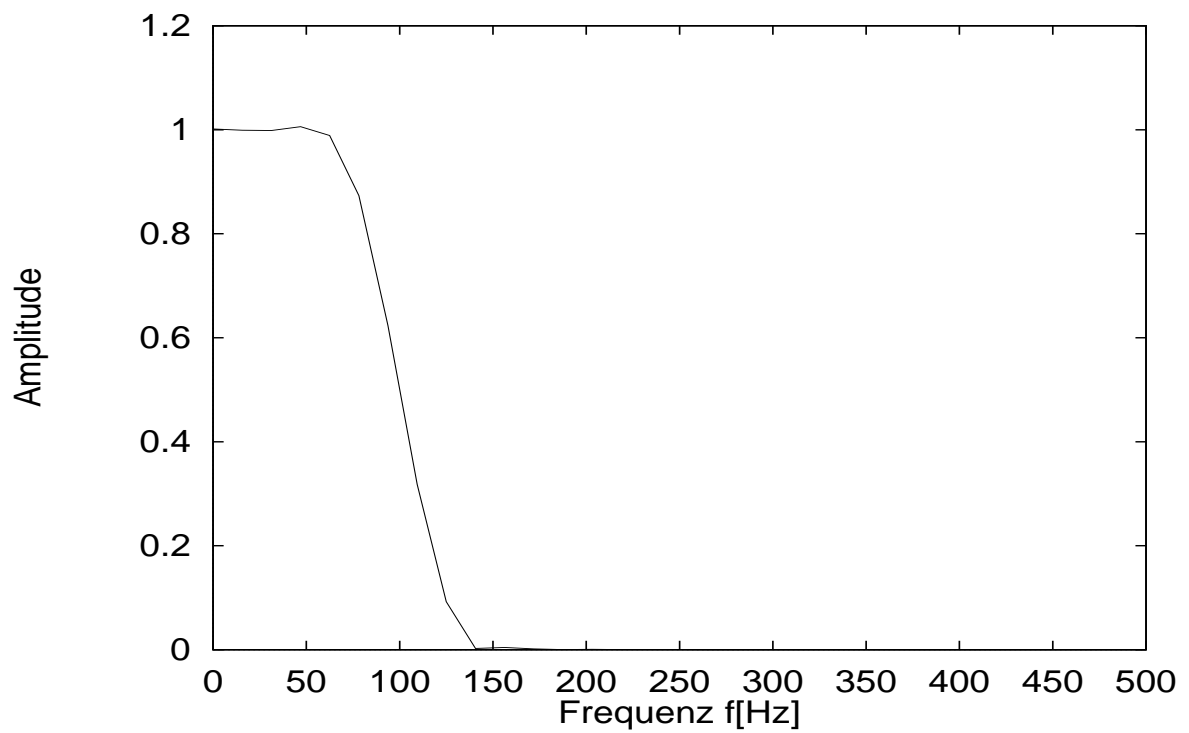
3) Betrag der Übertragungsfunktion für $dt = 2$ ms, $f_L = 100$ Hz und Cosinusfenster

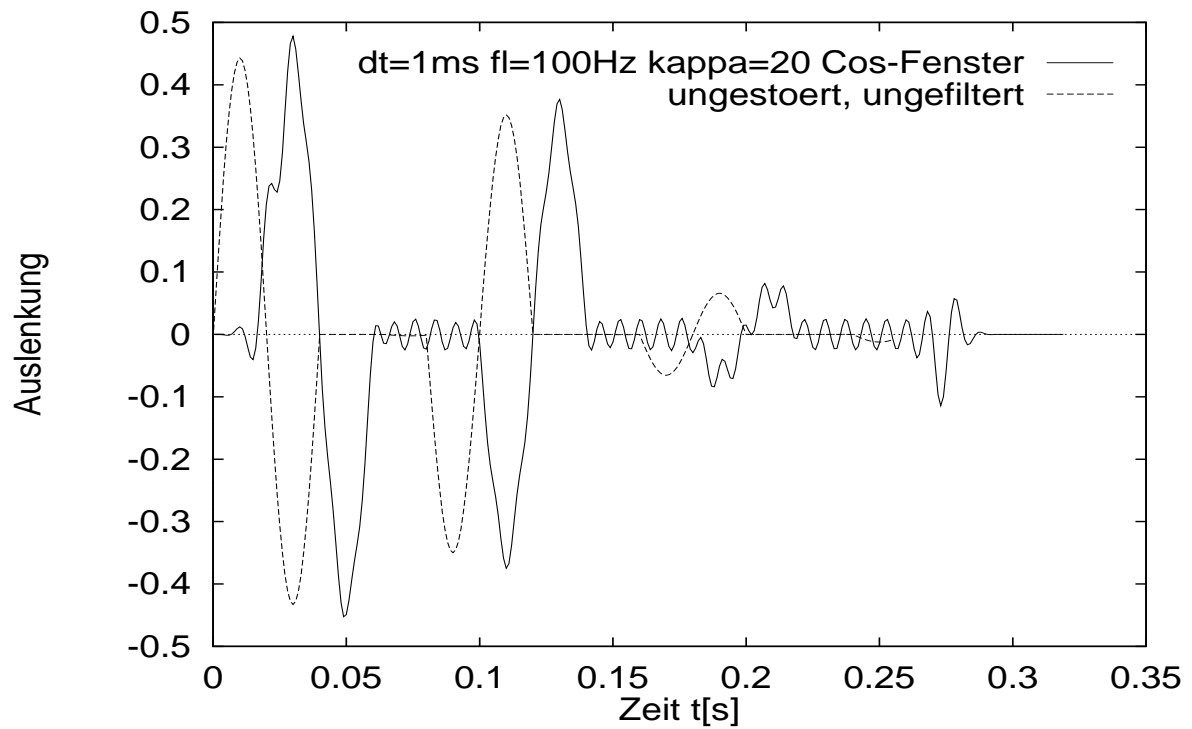


3) Betrag der Übertragungsfunktion für $dt = 2$ ms, $f_L = 100$ Hz und Cosinusfenster

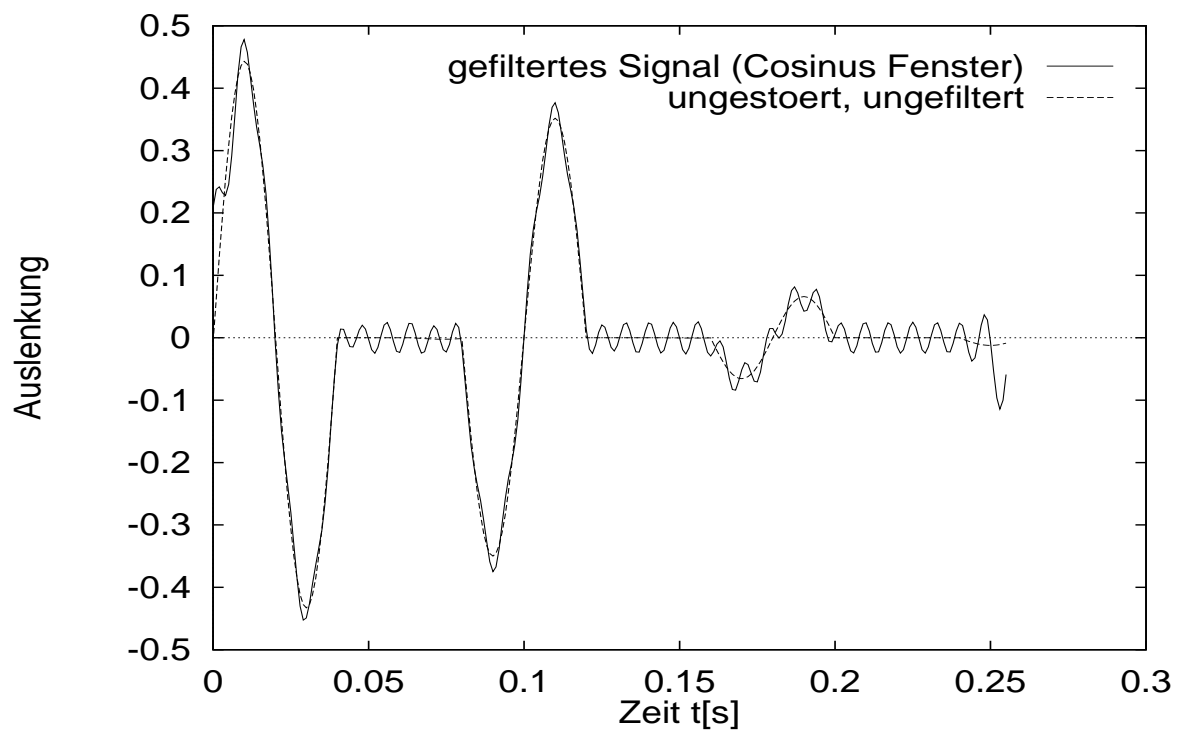


Gestörtes Seismogramm, Störfrequenz $f = 130$ Hz

4c) Filteroperator für $dt = 1\text{ms}$, $f_L = 100\text{ Hz}$, $\kappa = 20$ und Cosinus-FensterÜbertragungsfunktion für $dt = 1\text{ ms}$, $f_L = 100\text{ Hz}$, $\kappa = 20$ und Cosinus-Fenster



4d) Gefiltertes Seismogramm (zeitlich verschoben durch das Filtern)



4d) Gefiltertes Seismogramm (zeitliche Verschiebung korrigiert)

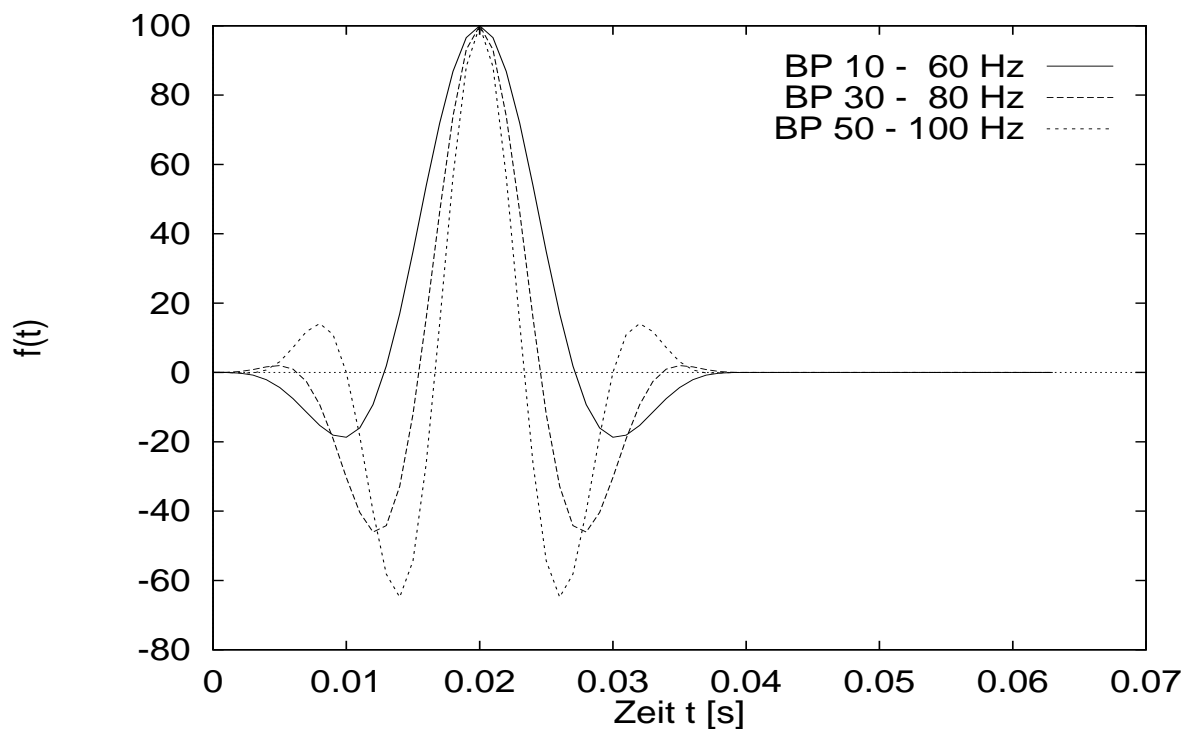
7 Bandpaßfilterung

7.1 Aufgabenstellung:

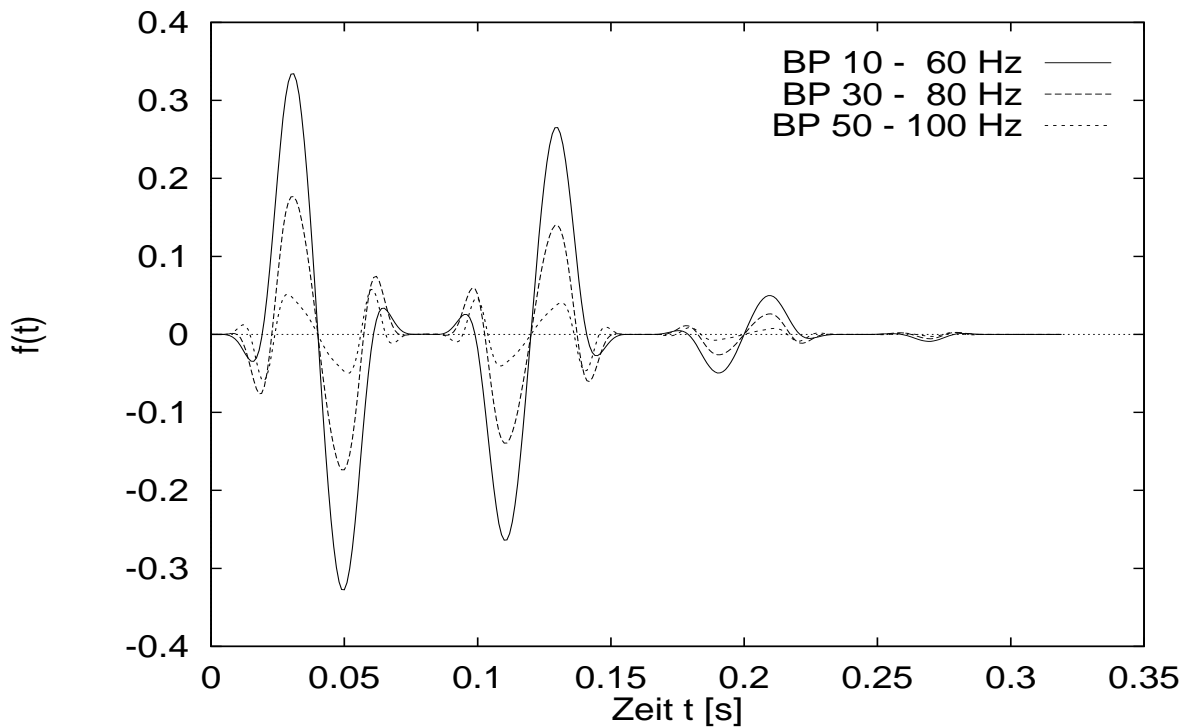
- 1) Erweiterung des Unterprogramms von Aufgabe 6, Teil 1), derart, daß man beliebige Bandpässe, einschließlich Tief- und Hochpaß, behandeln kann
- 2) Numerische Rechnungen für 3 Bandpässe (halber Filteroperator, Betrag der Übertragungsfunktion):

$$\Delta t = 0.001 \text{ s, Cosinus-Fenster, } \kappa = 20$$
 - a) $f_1 = 10 \text{ Hz, } f_2 = 60 \text{ Hz}$
 - b) $f_1 = 30 \text{ Hz, } f_2 = 80 \text{ Hz}$
 - c) $f_1 = 50 \text{ Hz, } f_2 = 100 \text{ Hz}$
- 3) Bandpaßfilterung von synthetischen Seismogrammen mit den Bandpässen aus Teil 2). Man nehme dasselbe Schichtenmodell wie in Aufgabe 6, Teil 4), und rechne mit $\Delta t = 0.001 \text{ s, } LX = 256$, aber $LT = 20$. Keine Überlagerung von Störschwingungen. Welche Resultate erwartet man? Wie groß ist die Akausalität der Reflexionen?

7.2 Plots für Aufgabe 7



2) Filterkoeffizienten verschiedener Bandpassfilter



3) Verschiedene Bandpassfilterungen des ungestörten Seismogramms

7.3 Akausalität

Die Impulsantwort der Filter zeigt, daß der Beitrag vor dem Maximum immer mehr anwächst je höher der Durchlaßbereich des Filters im Frequenzbereich ist. Daher nimmt auch die Akausalität zu.

(Da die Impulsantworten ihr Maximum bei $t = 0.02s$ haben, verschiebt sich das gefilterte Seismogramm auch um $t = 0.02s$.)

8 Butterworth-Tiefpaß n-ter Ordnung

8.1 Aufgabenstellung:

- 1) Man schreibe ein Unterprogramm zur Berechnung des Butterworth - Tiefpasses der Ordnung n . Eingabeparameter:

n : Ordnung
 Δt : Zeitintervall für die Impulsantwort (Filteroperator)
 LF : Länge des Filteroperator (Zweierpotenz)
 f_L : Grenzfrequenz ($= \omega_L/2\pi$):
 sollte höchstens gleich der halben Nyquistfrequenz sein

Geplottet werden sollen der Filteroperator $f(t)$ und Betrag und Phase der Übertragungsfunktion $\bar{f}(\omega)$ für:

$n = 1, 2, \dots, 8$, $\Delta t = 1$ s, $LF = 64$ und $f_L =$ halbe Nyquistfrequenz .

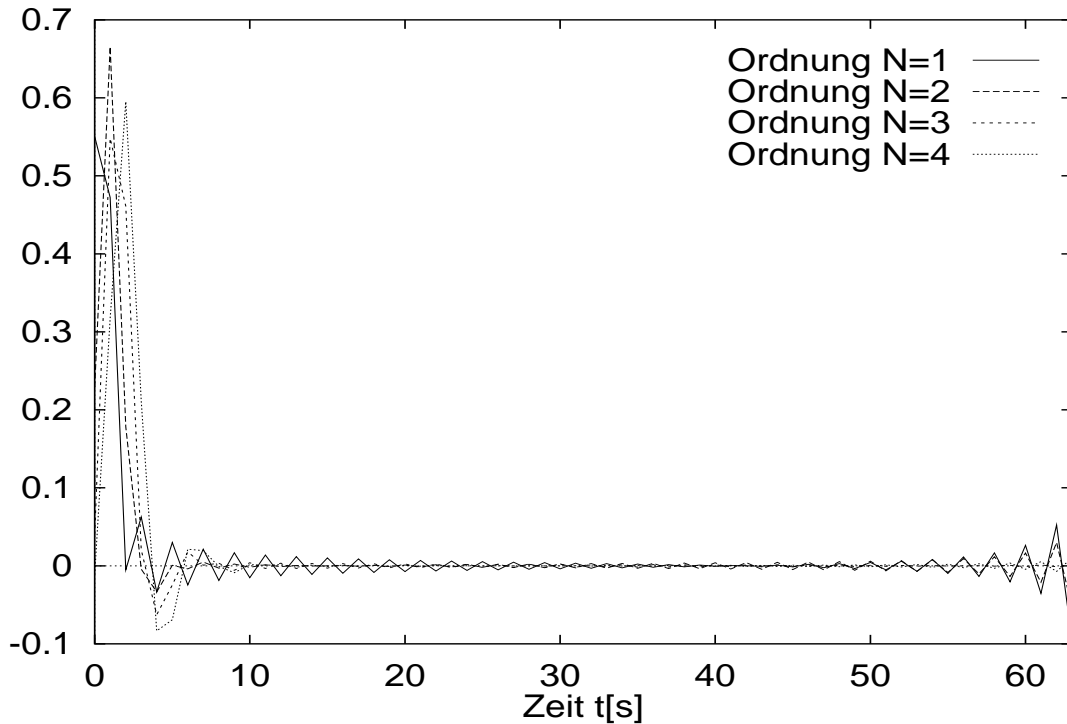
Vor der Rücktransformation von $\bar{f}(\omega)$ ($\rightarrow f(t)$) in den Zeitbereich mit *FORK* bzw. *four1()* multipliziert man mit dem aus (2-27) folgenden Faktor $1/(\sqrt{LF}\Delta t)$. Dann muß man zur Tiefpaßfilterung nur noch mit $f(t)$ falten, ohne einen weiteren Faktor anzubringen.

Die Summe der Filterkoeffizienten muß in diesem Fall gleich $1/\Delta t$ sein (warum?); das ist eine Kontrollmöglichkeit.

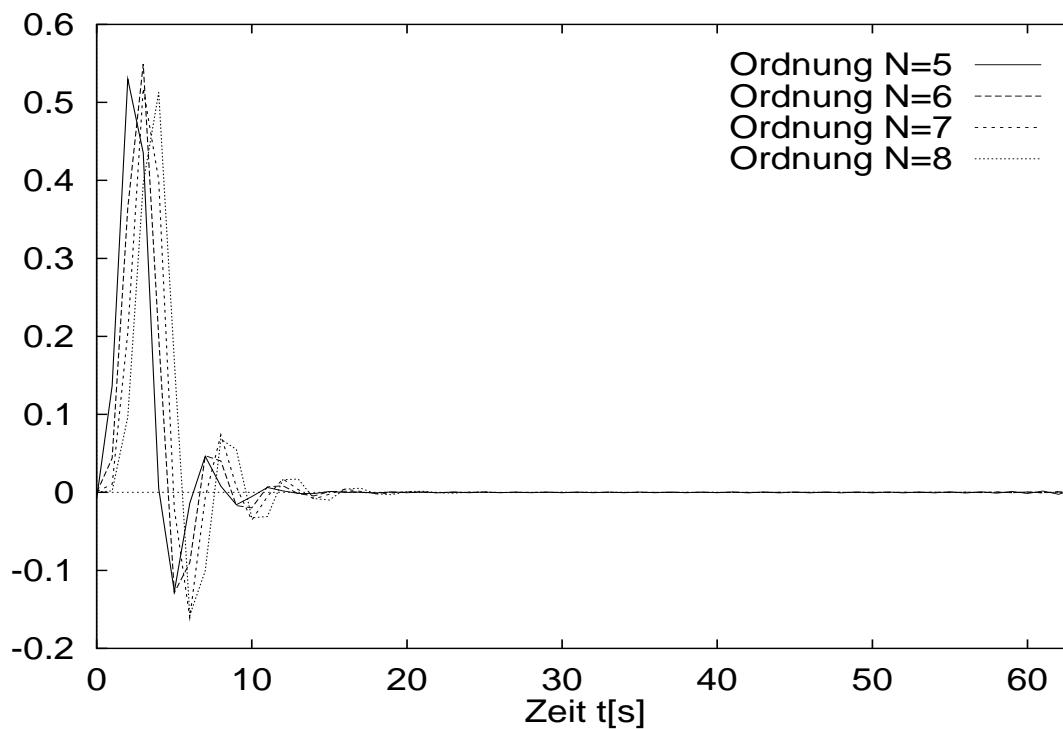
2 Man wiederhole die Tiefpaßfilterung von Aufgabe6, Teil 4), mit einem Butterworth-Filter, dessen Parameter wie folgt lauten:

$n = 8$, $\Delta t = 0.001$ s, $LF = 64$, $f_L = 100$ Hz.

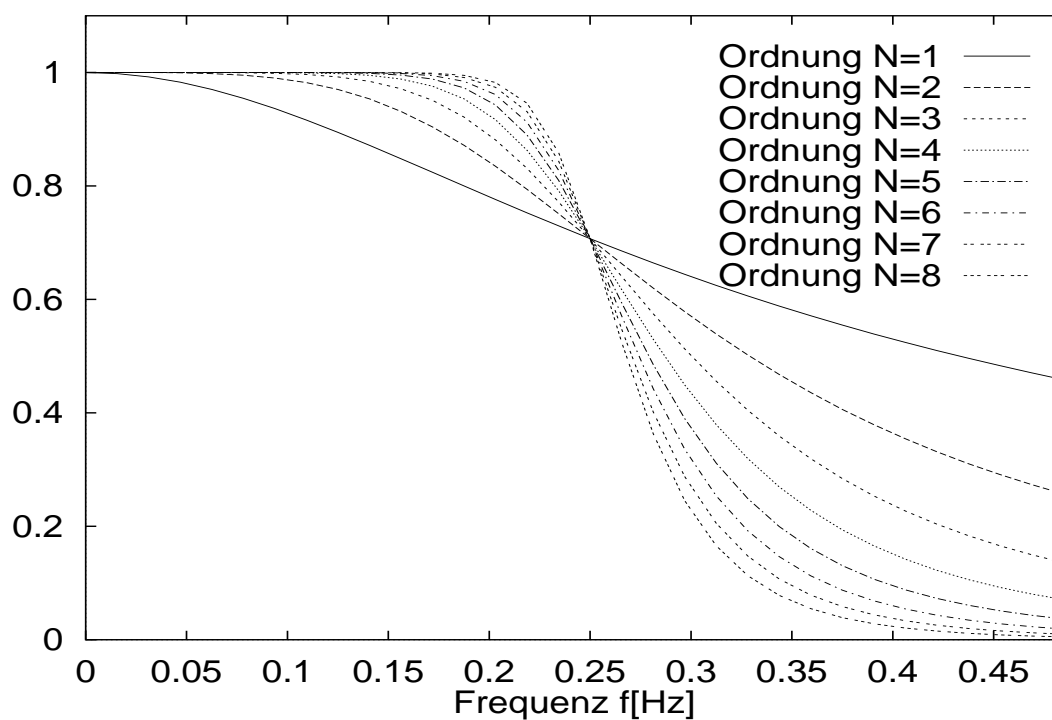
8.2 Plots für Aufgabe 8:



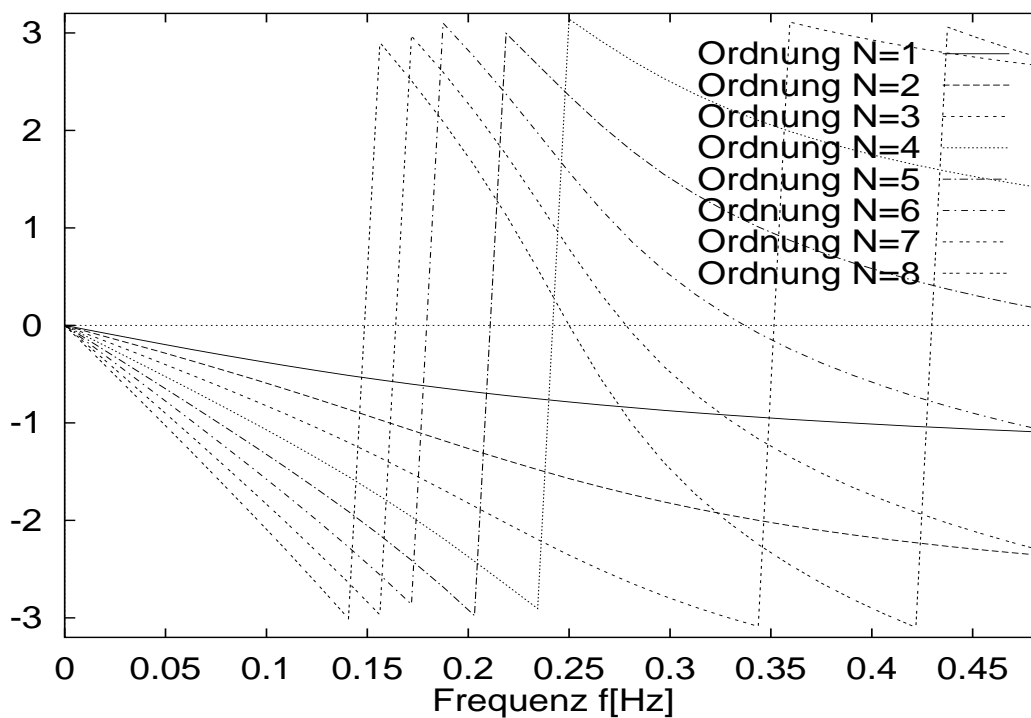
Filterkoeffizienten verschiedener Butterworthfilter



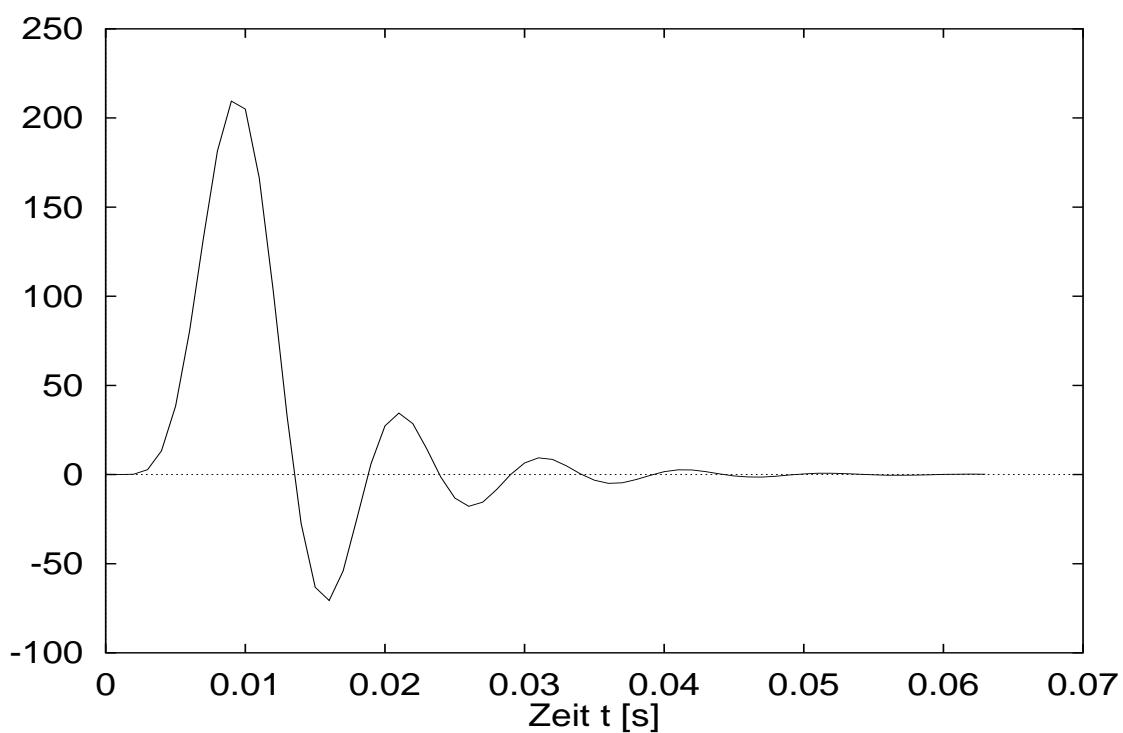
Filterkoeffizienten verschiedener Butterworthfilter



Betrag der Übertragungsfunktion verschiedener Butterworthfilter

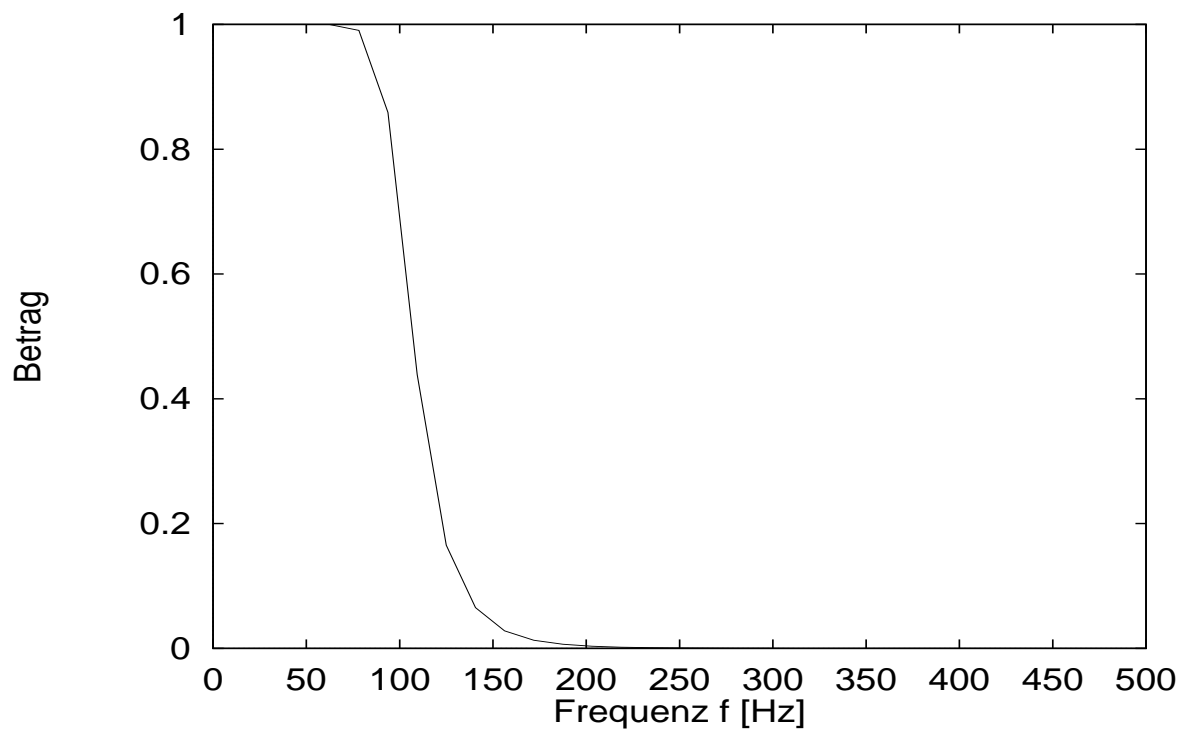


Phase der Übertragungsfunktion verschiedener Butterworthfilter

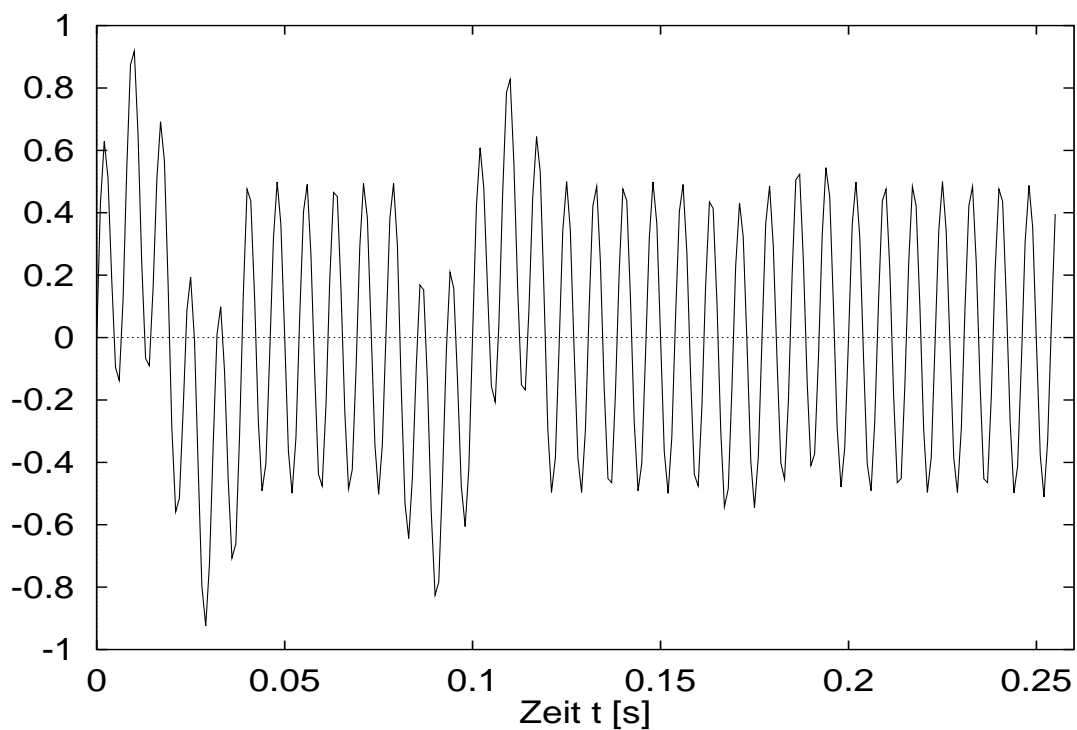


Filterkoeffizienten

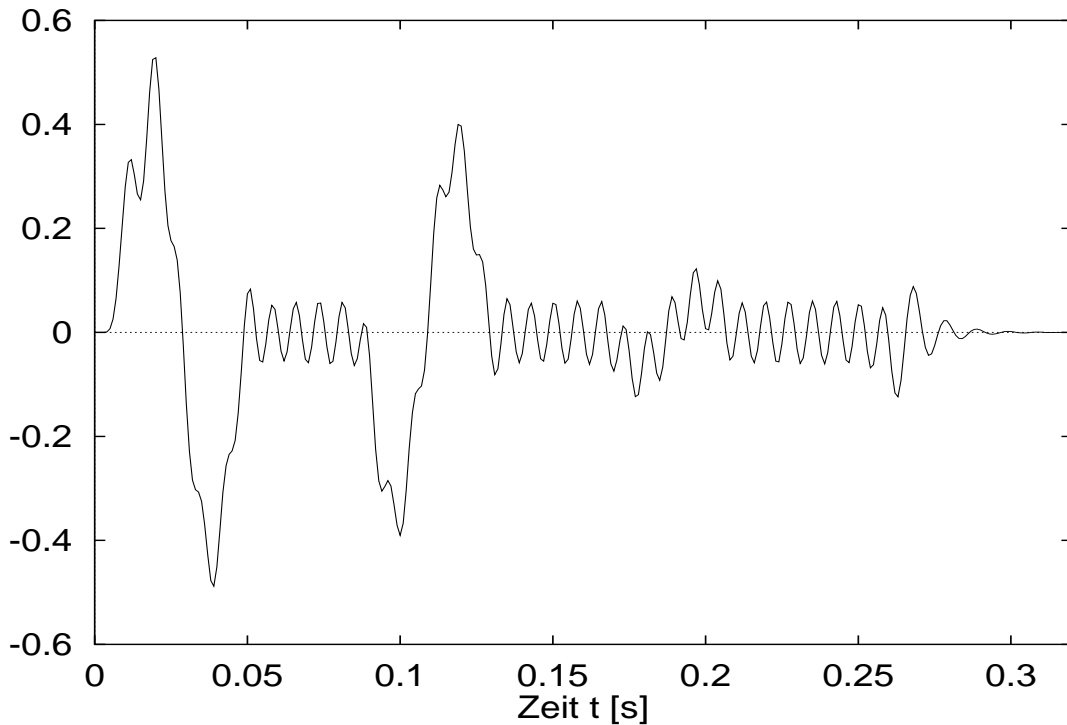
Butterworthfilter mit $n = 8$, $dt = 1$ ms, $f_L = 100$ Hz



Übertragungsfunktion
Butterworthfilter mit $n = 8$, $dt = 1$ ms, $f_L = 100$ Hz



Gestörtes Seismogramm

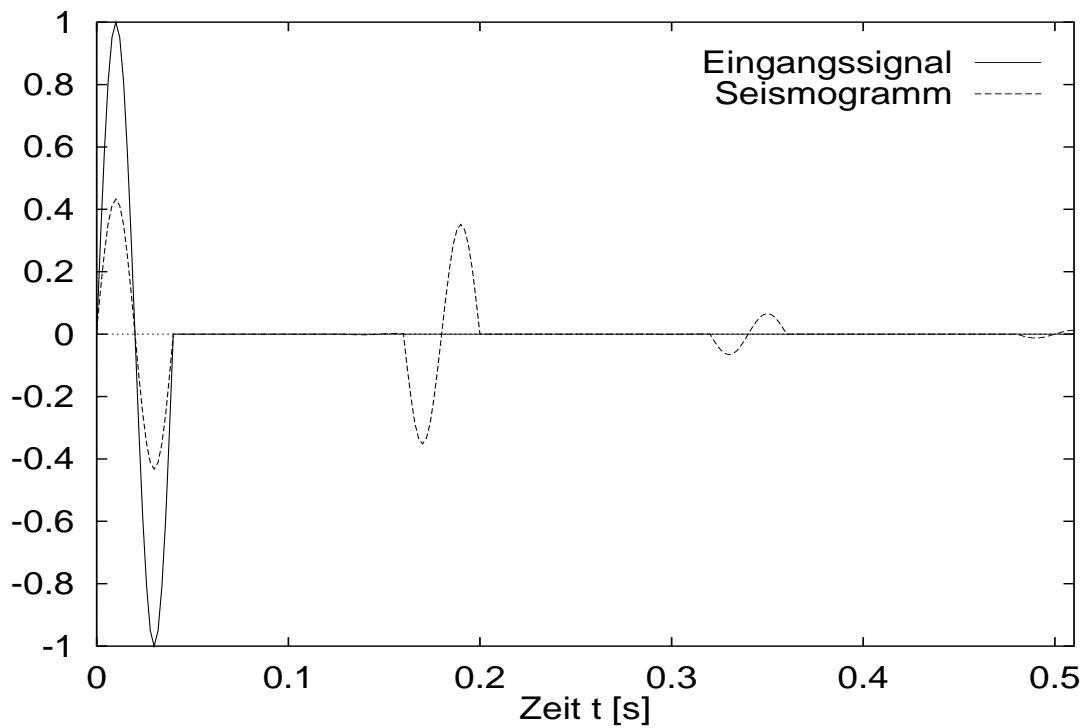


Gefiltertes Seismogramm, Butterworth mit $n = 8$, $dt = 1$ ms, $f_L = 100$ Hz

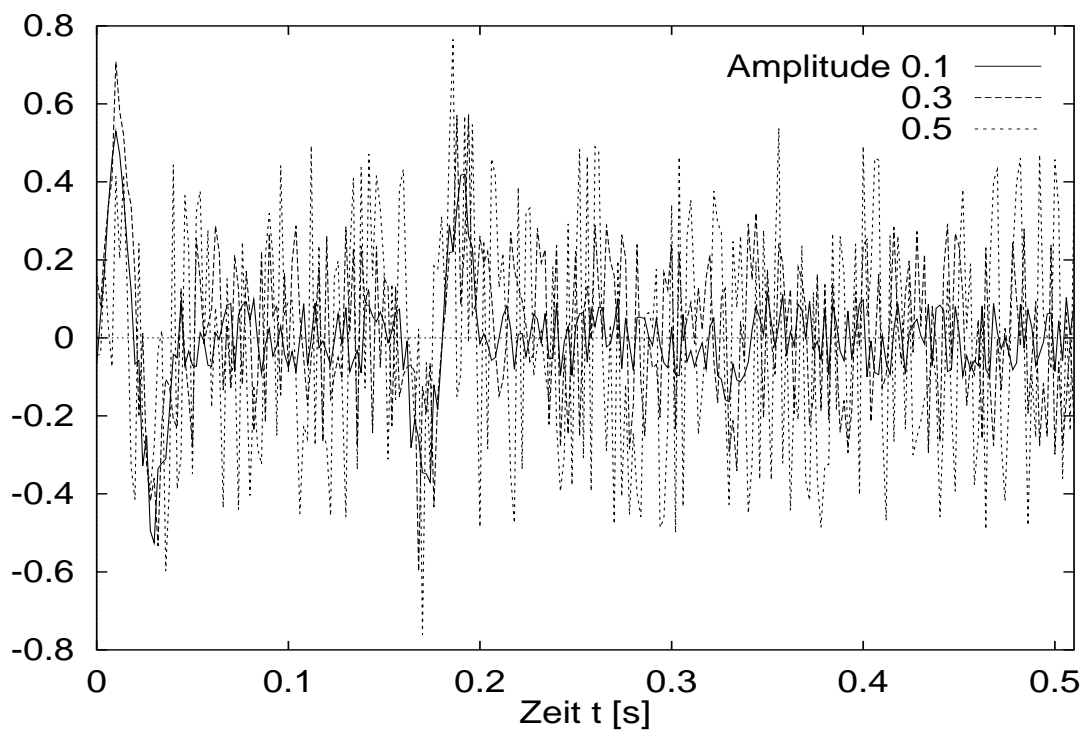
9 Kreuzkorrelation

9.1 Aufgabenstellung:

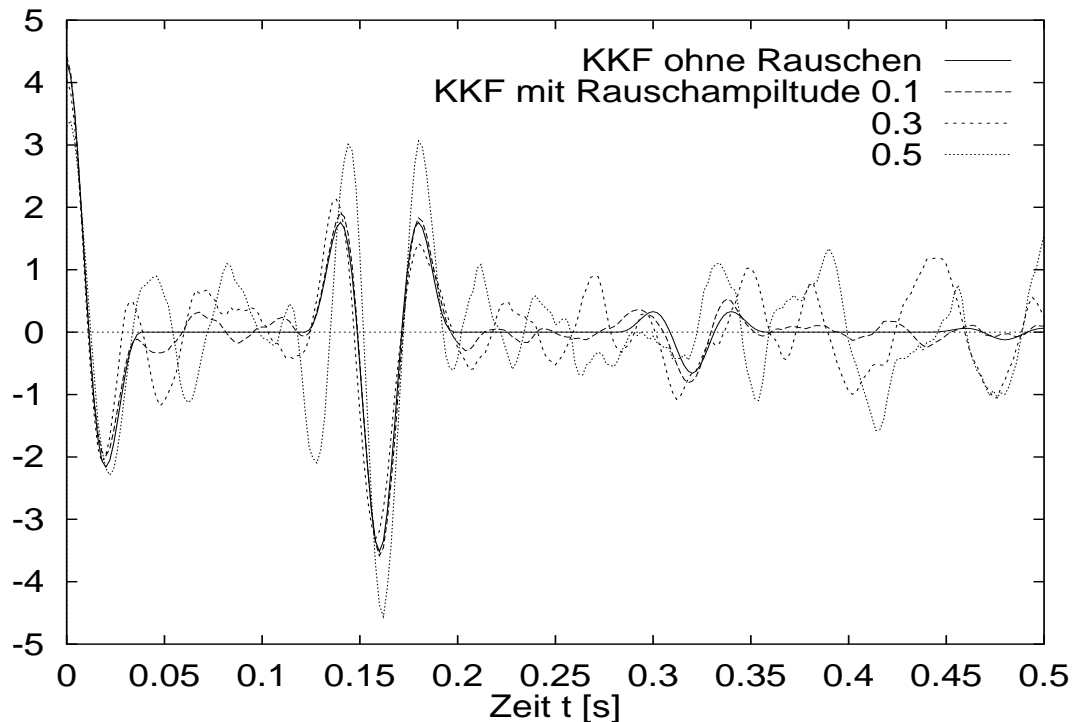
- 1) Man schreibe ein Unterprogramm für die Berechnung der KKF $\Phi_{sX}(t)$ zweier Zeitfunktionen $x(t)$ (Länge LX) und $s(t)$ (Länge $LS < LX$) für $t \geq 0$ (vgl. S.83 im Skript). Die KKF soll für $LCOR$ Punkte berechnet werden ($LCOR \leq LX - LS$).
- 2) Berechnung des synthetischen Seismogramms für das Standard - Schichtenmodell von Aufgabe 3 ($\Delta t = 0.002$ s, $LX = 256$, $LT = 20$, Dicke der zweiten Schicht 200 m statt 100 m) und Korrelation mit dem Eingangssignal.
- 3) Addition von künstlich erzeugtem Rauschen $n(t)$ zum Seismogramm von Teil 2) und erneute Korrelation. Dabei Verwendung von Pseudo - Zufallszahlen, z.B. mit der Routine im Skript, S. 84f oder in C mit $rand()$ und $srand()$.

9.2 Plots für Aufgabe 9:

Eingangssignal und Seismogramm
(letzteres um die Laufzeit des 1. Echos verschoben)



Ausgangssignal überlagert mit Rauschen



KKF von Ausgangssignal und verrauschtem Ausgangssignal

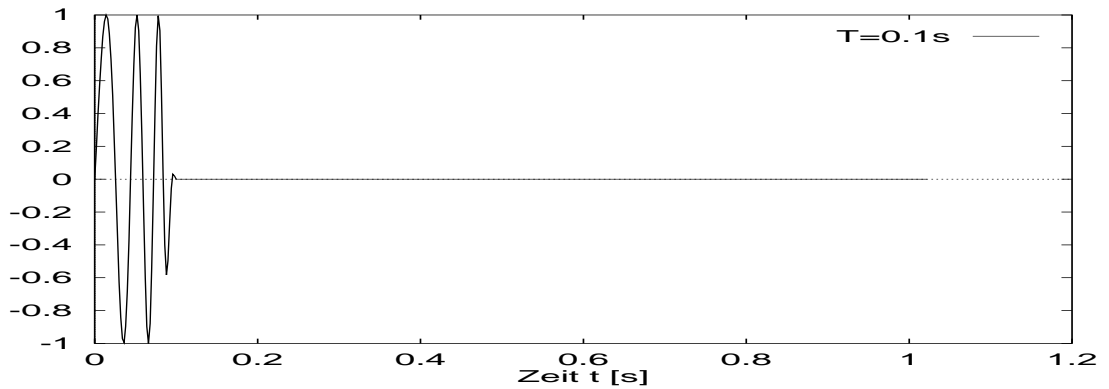
10 Vibroseis

10.1 Aufgabenstellung:

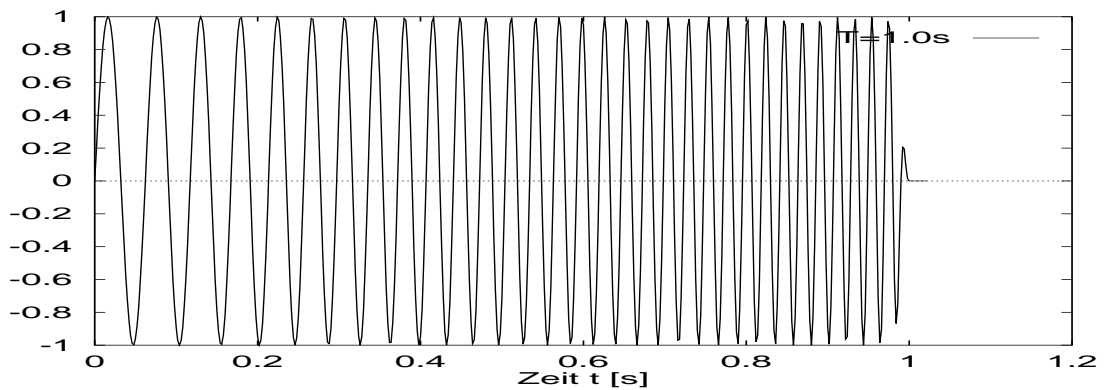
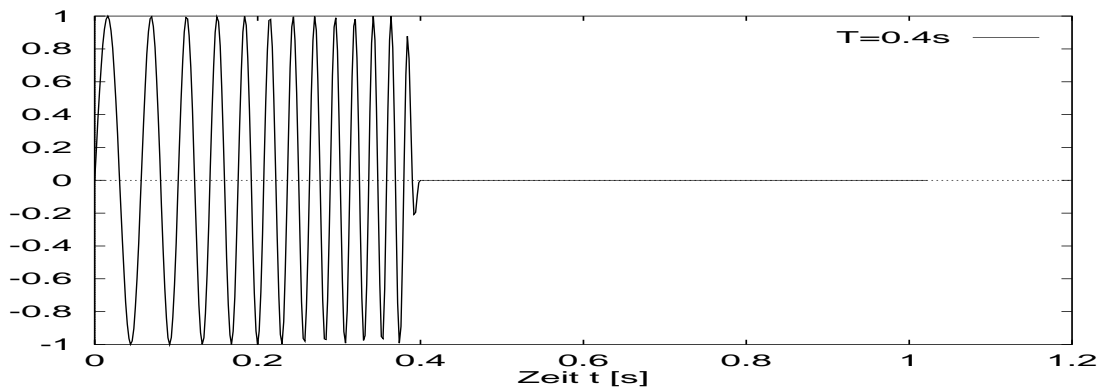
- 1) Man schreibe ein Unterprogramm zur Berechnung des Sweeps (6.9) zu vorgegebenen Werten von ν_1 , ν_2 , T und Δt . Damit $s(t)$ bei $t = T$ keinen Sprung macht, lege man über die letzten 10 Punkte ein Cosinus-Fenster. Auffüllen mit Nullen bis zur Zweierpotenz von Stützstellen.
- 2) Man erzeuge:
 - a) Plots der Sweeps für $\nu_1 = 15$ Hz, $\nu_2 = 50$ Hz, $\Delta t = 0.002$ s und $T = 0.1$, 0.4 und 1.0 s
 - b) Amplitudenspektren der Sweeps
 - c) AKFs der Sweeps mit dem Korrelationsprogramm von Aufgabe 9 für $LCOR = 60$ ($x(t) = \text{Sweep mit 60 Nullen}$, $s(t) = \text{Sweep}$).
Wodurch unterscheiden sich die AKFs vor allem?
- 3) Berechnung des synthetischen Seismogramms wie in Aufgabe 9, aber für einen Sweep als Eingangssignal

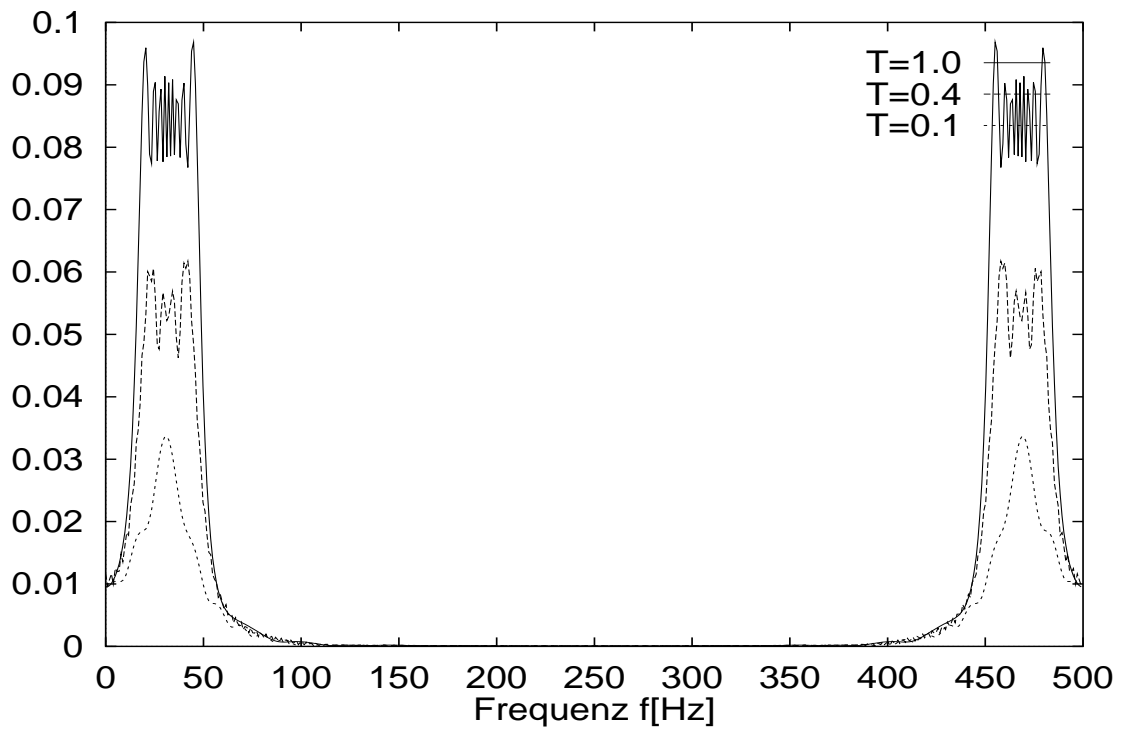
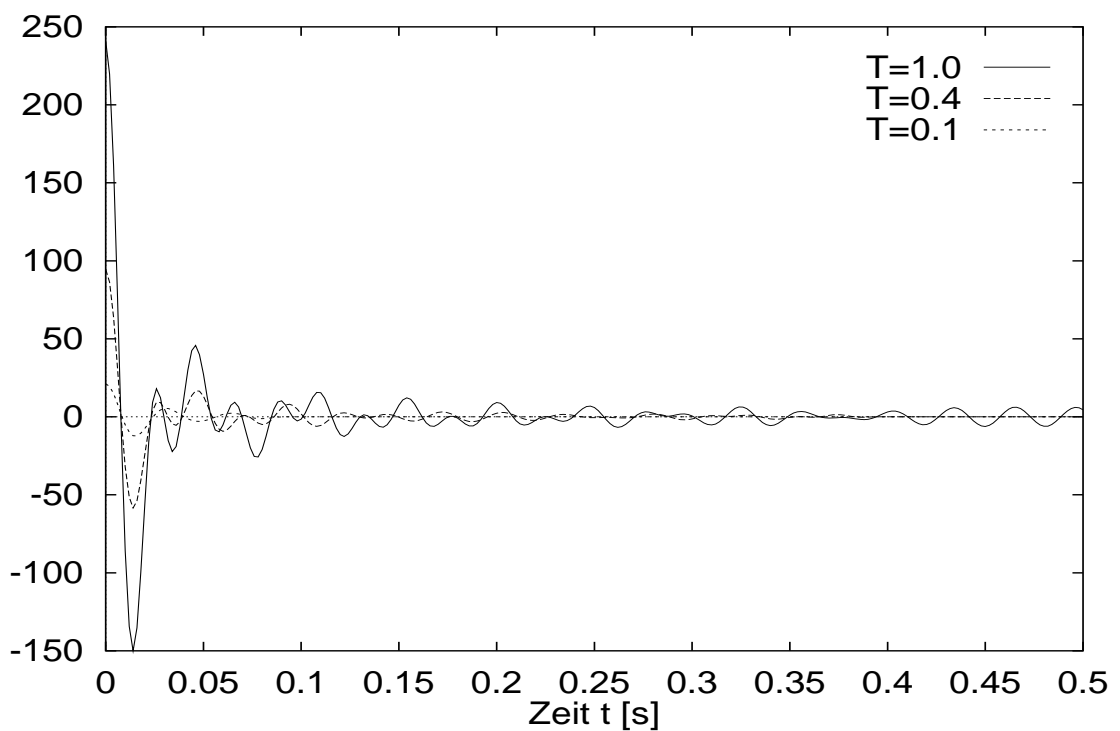
($\nu_1 = 15$ Hz, $\nu_2 = 50$ Hz, $T = 0.4$ s, $\Delta t = 0.002$ s)
 und für $LX = 512$ Punkte. Dann korreliere man Seismogramm
 ($\rightarrow x(t)$) und Sweep ($\rightarrow s(t)$, $LS = 200$): $LCOR = LX - LS$.

10.2 Plots für Aufgabe 10:

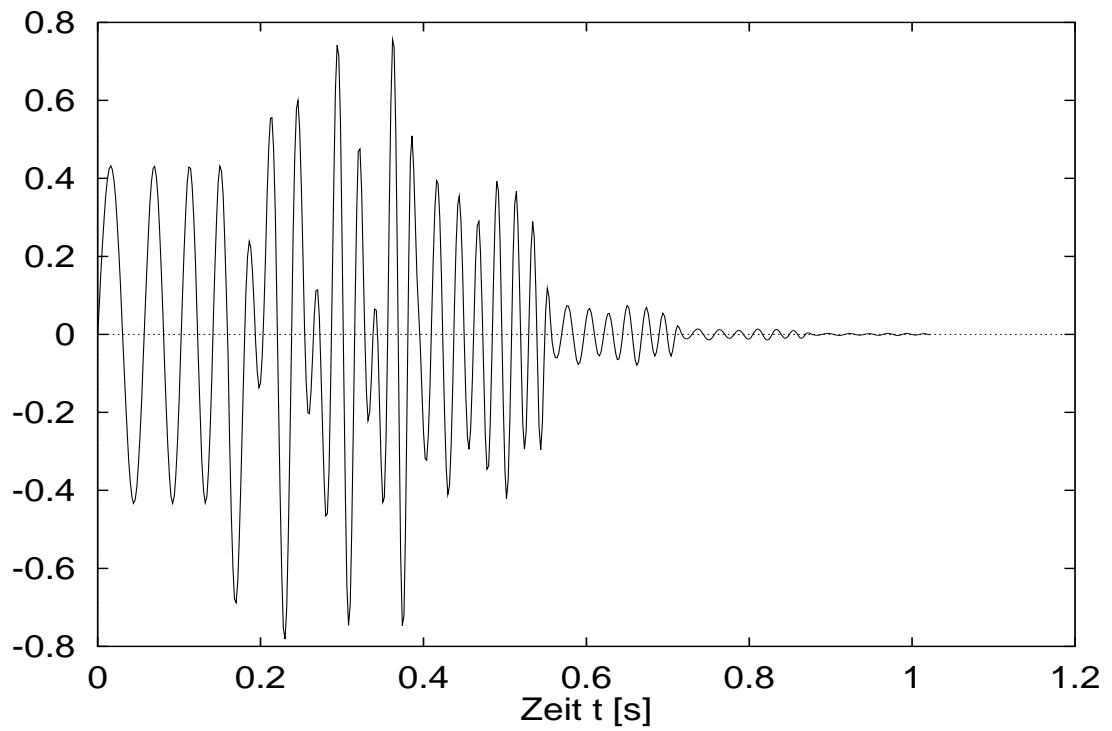
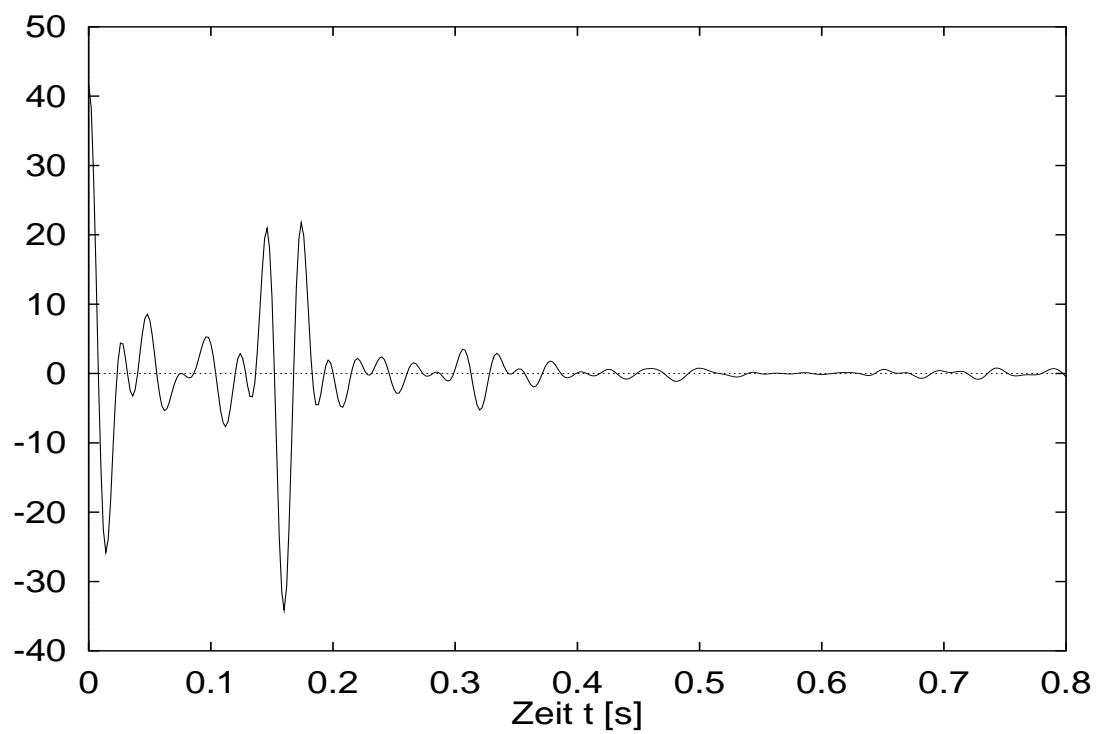


2a) Sweeps für verschiedene T's



Amplitudenspektren der Sweeps mit $T = 0.1, 0.4$ und 1.0 s

AKF's der Sweeps

Synthetisches Seismogramm des Sweeps mit $T = 0.4$ sKKF des Seismogramms ($T = 0.4$ s) mit dem Sweep

11 Rekursives Notch-Filter

11.1 Aufgabenstellung:

- 1) Man berechne über $F(z)$ aus (8-5) die Übertragungsfunktion $\bar{f}(\omega)$ und die Impulsantwort $f(t)$ eines Notch-Filters mit der Sperrfrequenz $\omega_0 = 1/4\omega_N$ und $\epsilon = 0.1$ bzw. 0.01 .
Die Länge von $f(t)$ sei 128 und Δt z.B. 0.002 s.

- 2) Man schreibe ein FORTRAN / C Unterprogramm $REKURS(X, Z, LX, A, LA, B, LB)$ zur rekursiven Filterung gemäß (8-2):

X = Filtereingang

Z = Filterausgang

A = Zählerkoeffizienten (Znzahl $LA = k + 1$)

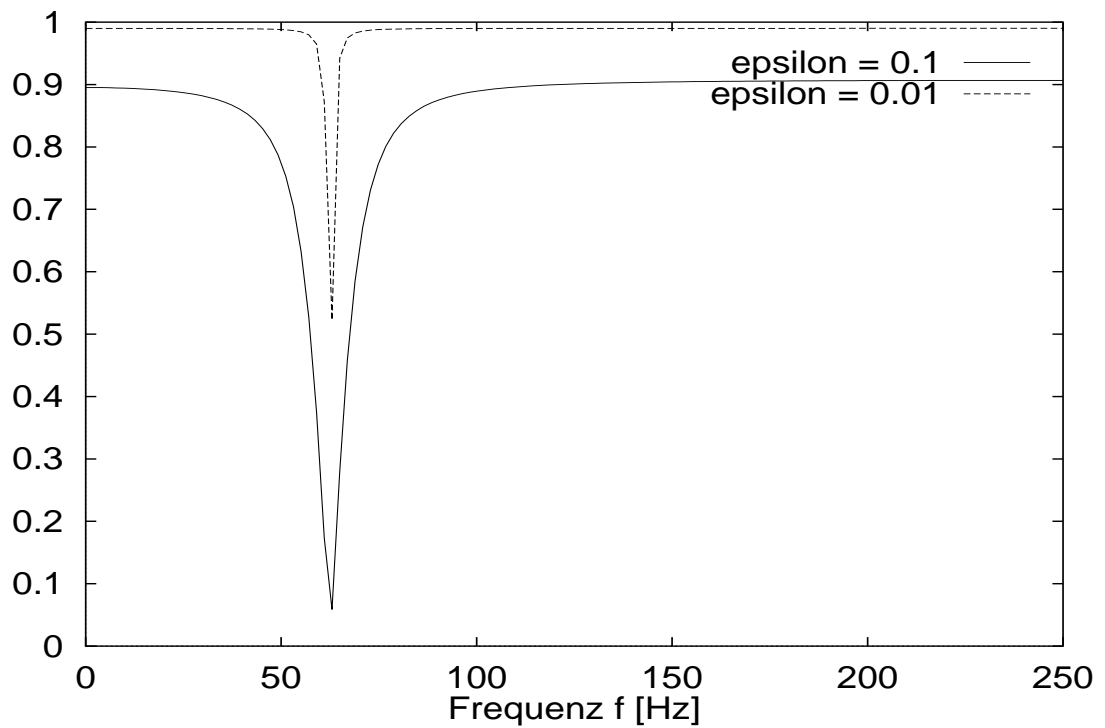
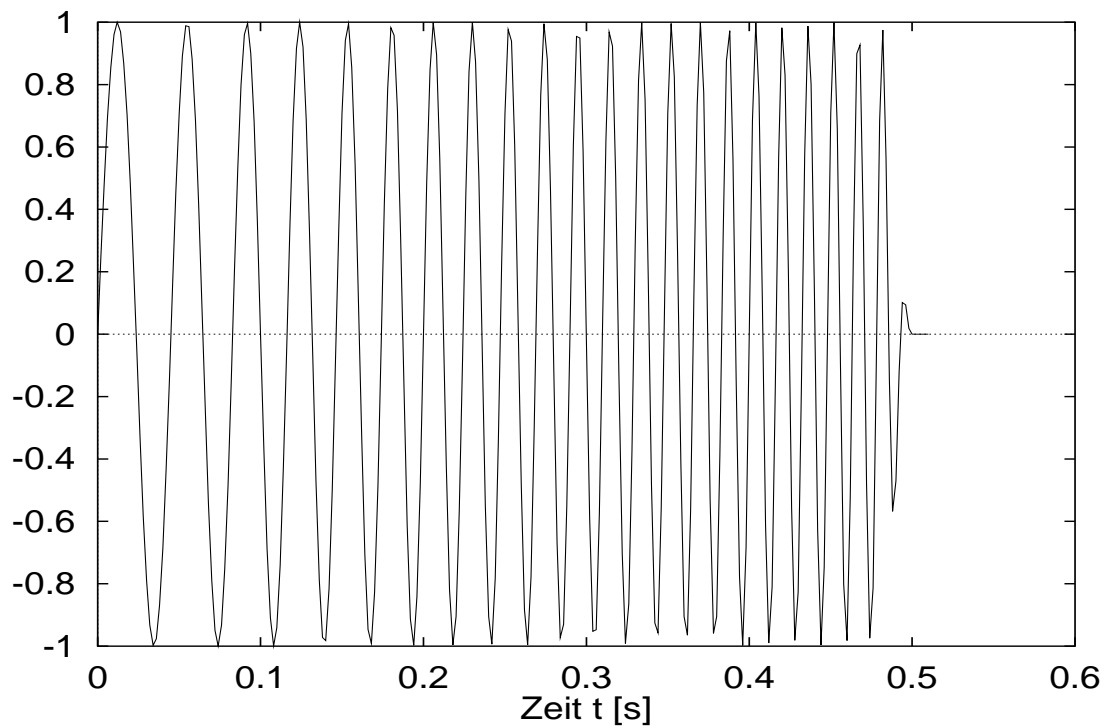
B = Nennerkoeffizienten (Anzahl $LB = l + 1, B(1) = 1$)

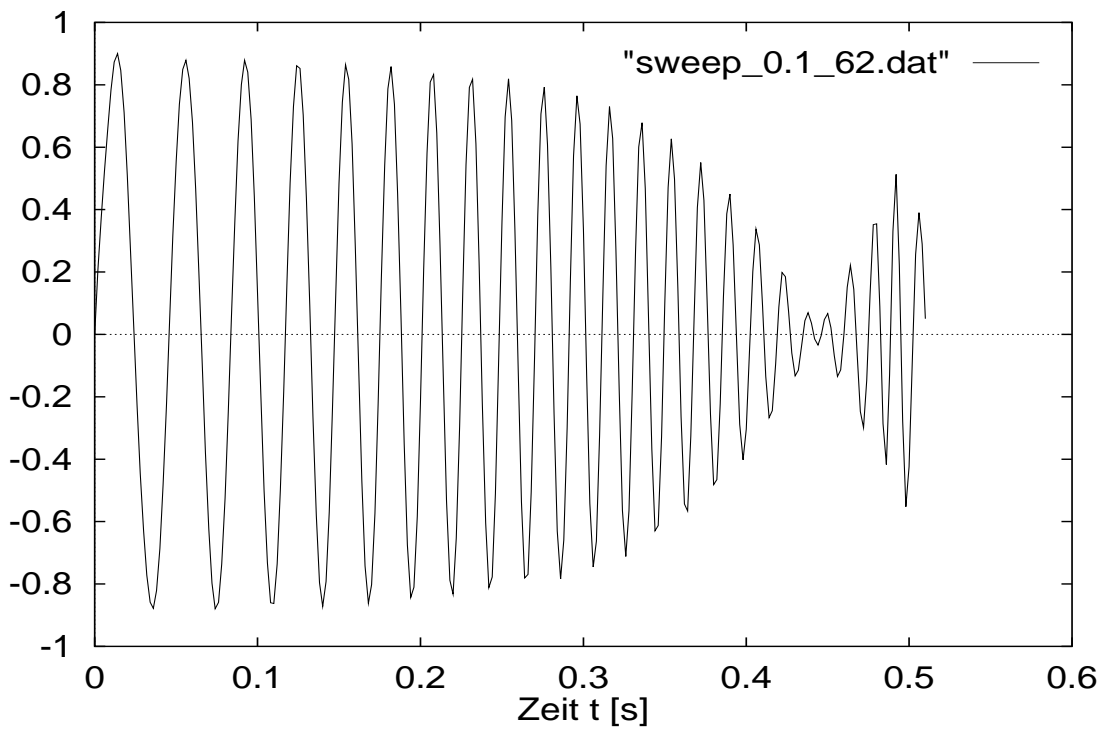
- 3) Rekursive 40 Hz -Notch - Filterung des Sweeps

$$s(t) = \sin\left(2\pi\left(\nu_1 + \frac{\nu_2 - \nu_1}{2T}t\right)t\right)$$

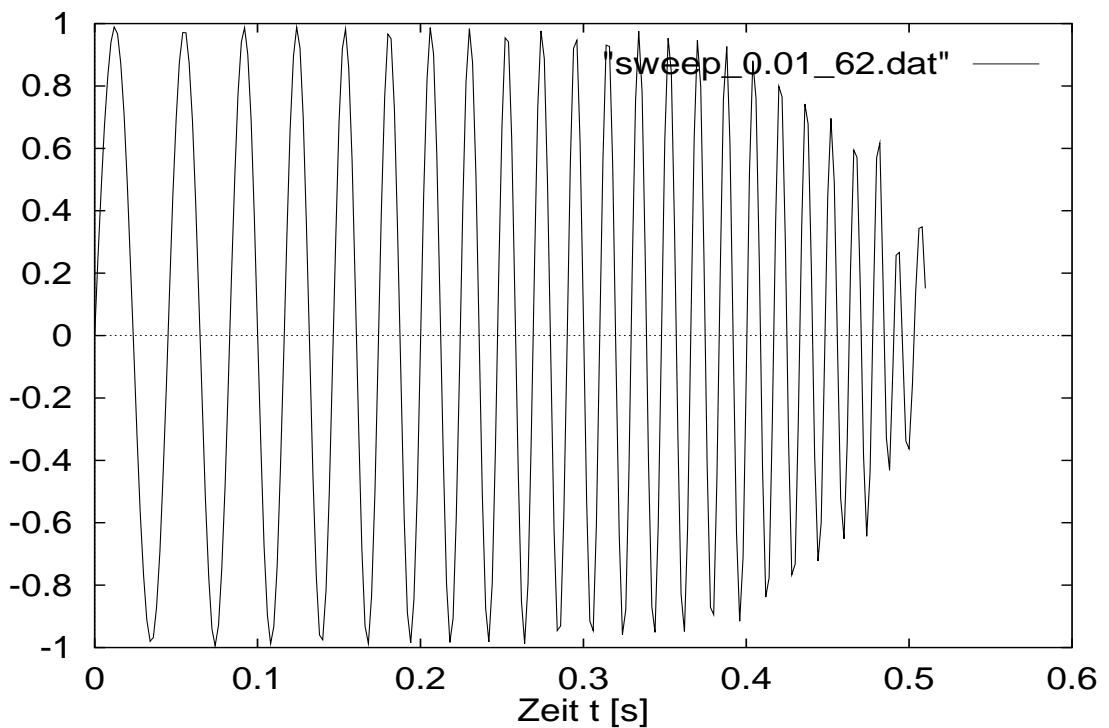
mit den Grenzfrequenzen $\nu_1 = 20$ Hz und $\nu_2 = 70$ Hz und der Dauer $T = 0.5$ s. Weitere Parameter sind: Zeitintervall $\Delta t = 0.002$ s, $\epsilon = 0.1$ bzw. 0.01 . Kann man ϵ beliebig klein machen?

11.2 Plots für Aufgabe 11:

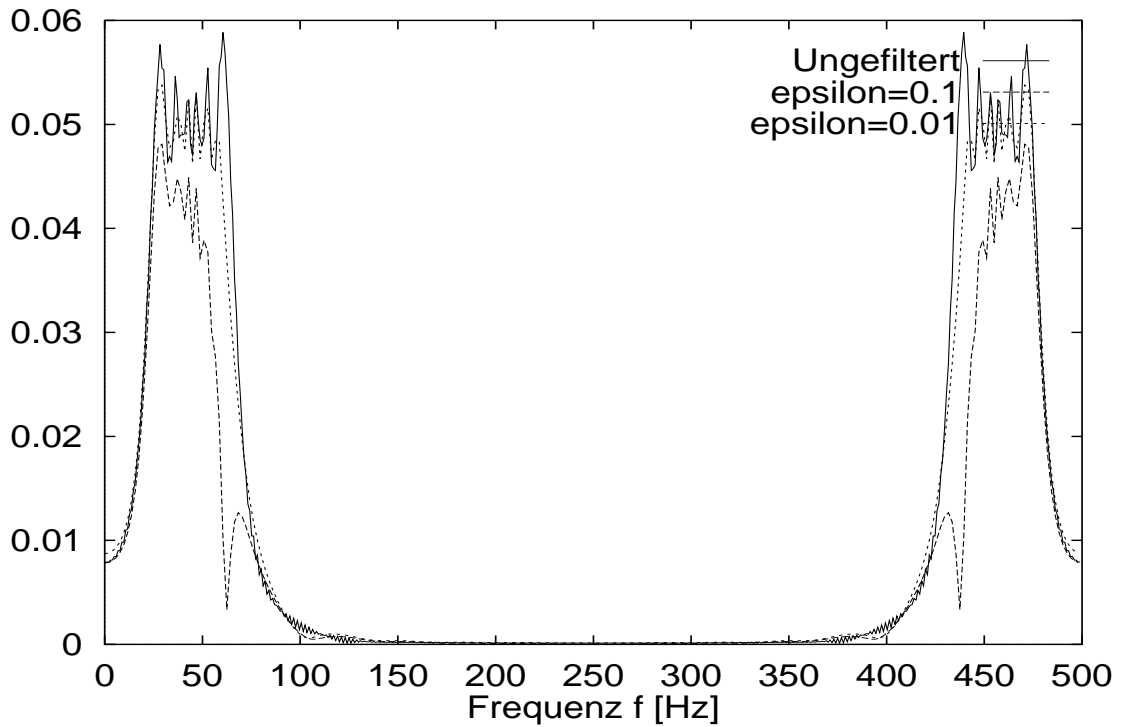
1) Amplitudenspektrum eines Notch-Filters mit $f_L = 62.5$ Hz3) Sweep mit $f = 20 - 70$ Hz, $T = 0.5$ s



Sweep mit $f = 20 - 70$ Hz, $T = 0.5$ s
gefiltert mit Notch-Filter, $\epsilon = 0.1$, $f = 62.5$ Hz



Sweep mit $f = 20 - 70$ Hz, $T = 0.5$ s
gefiltert mit Notch-Filter, $\epsilon = 0.01$, $f = 62.5$ Hz



Amplitudenspektrum des Sweeps

12 Rekursiver Butterworth-Tiefpaß

12.1 Aufgabenstellung:

- 1) Man berechne die z-Transformierte des Butterworth-Tiefpasses 2. Ordnung, vgl. (5-27) auf S.65 im Skript,

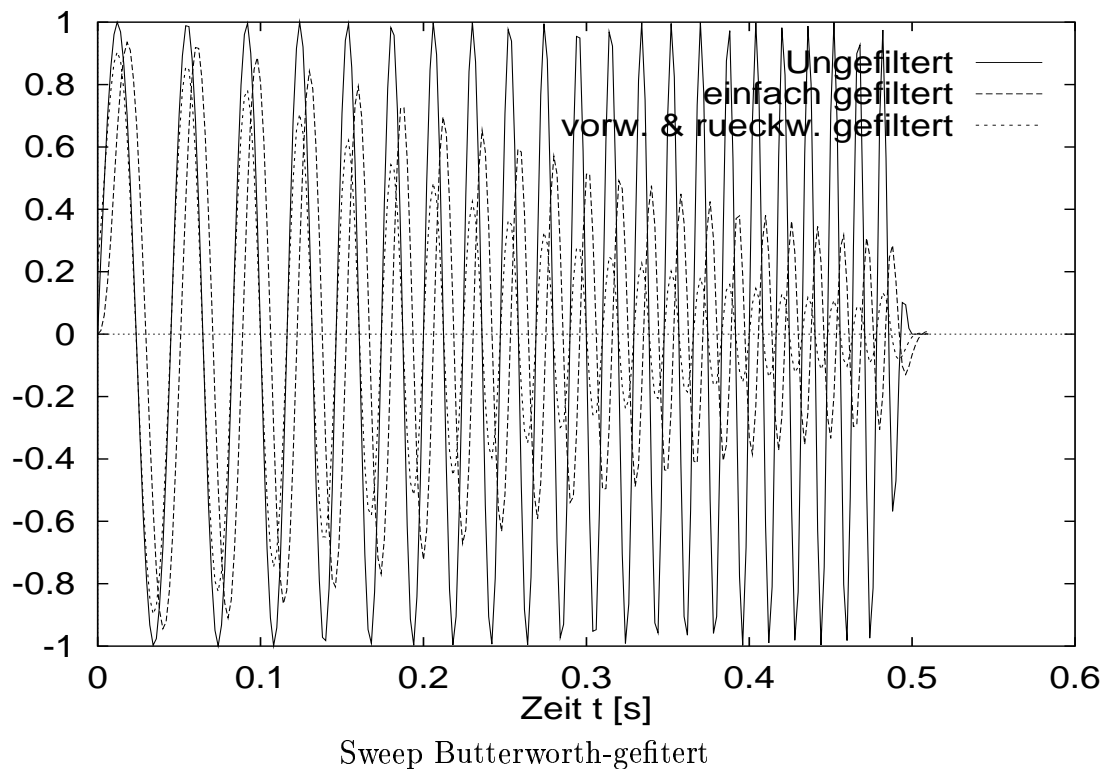
$$\bar{f}(\omega) = \frac{-1}{\left(\frac{\omega}{\omega_L} - e^{i\pi/4}\right)\left(\frac{\omega}{\omega_L} - e^{i3\pi/4}\right)}$$

mit der Grenzfrequenz ω_L . Es ergibt sich:

$$F(z) = \frac{\Omega_L^2 (z+1)^2}{\Omega_L^2 + \frac{2\sqrt{2}\Omega_L}{\Delta t} + \frac{4}{\Delta t^2} + (2\Omega_L^2 - \frac{8}{\Delta t^2})z + (\Omega_L^2 - \frac{2\sqrt{2}\Omega_L}{\Delta t} + \frac{4}{\Delta t^2})z^2}$$

- 2) Man bestimme die Koeffizienten des entsprechenden rekursiven Filters und filtere den Sweep von Aufgabe 11, 3) für $\omega_L/2\pi = 40$ Hz.
- 3) Man führe die Tiefpaßfilterung des Sweeps vorwärts **und** rückwärts durch (vgl. Abschnitt 6.5. im Skript) und vergleiche mit dem Resultat von 2) und mit dem ungefilterten Sweep.

12.2 Plots für Aufgabe 12:



13 Analyse und Invertierung eines Wavelets

13.1 Aufgabenstellung:

- 1) Man zerlege das Wavelet (18, 45, 34, 8) in 2-Punkt-Wavelets durch Faktorenerlegung der z-Transformierten und zeige, daß es gemischtphasig ist. Man gebe das zugehörige minimalphasige Wavelet an und vergleiche den Aufbau der Energie in beiden Wavelets. Man skizziere die Abbildung des Einheitskreises durch die z-Transformierte in beiden Fällen (vgl. S. 126 im Skript).
- 2) Man invertiere beide Wavelets mit der auf S. 130 im Skript unten angegebenen ersten Methode (ca. 20 Punkte) und schreibe dazu ein Programm.

13.2 Analytische Rechnungen

- 1) minimalphasiges Wavelet
Das Ausgangswavelet lautet: (18, 45, 34, 8)

Zugehörige Z-Transformierte : $8z^3 + 34z^2 + 45z + 18$

Polynomdivision durch Nullstelle $z_1 = -2$:
 $(8z^3 + 34z^2 + 45z + 18)/(z + 2) = 8z^2 + 18z + 9$

Restliche Nullstellen: $z_2 = -3/4$ $z_3 = -3/2$

Die Z-Transformierte läßt sich damit schreiben als:
 $8z^3 + 34z^2 + 45z + 18 = (2 + z)(3 + 2z)(3 + 4z)$

Die 2-Punkt-Wavelets sind: $(2, 1)$; $(3, 2)$; $(3, 4)$.

Die Nullstellen z_1 und z_3 liegen außerhalb des Einheitskreises,
 z_2 liegt innerhalb. Das Wavelet ist also gemischtphasig.

Faltung der 2-Punkt-Wavelets liefert das minimalphasige Wavelet:
 $(24, 46, 29, 6)$

2) Energie

Die Energie des Wavelets zur Zeit $i \cdot \Delta t$ ist hier die Summe der Quadrate aller Samples bis zur Zeit $i \cdot \Delta t$.

Energie des Ausgangswavelets : $(324, 2349, 3505, 3569)$

Energie des minimalphasigen Wavelets : $(576, 2692, 3533, 3569)$

Die Energie des minimalphasigen Wavelets hat sich im Vergleich zum Ausgangswavelet nach vorne verschoben.

13.3 Listen und Fehlerenergieen

Teil 2)

Inputwavelet:	{1.0, 0.5}
Zielwavelet:	{1.0, 0.0}
AKF von Inputwavelet:	{1.25, 0.5}
KKF von Input mit Ziel:	{1.0, 0.0}
Inverser Filter zum Input:	{0.95238096, -0.38095242}
Faltung Invers*Input :	{0.95238096, 0.09523806 - 0.19047621}
<hr/> <hr/> Fehlerenergie:	<hr/> <hr/> $f = 0.047619$

Teil 3)

Wavelets und KKF's:

	minimalphasig	gemischtphasig	maximalphasig
Inputwavelet	{24,46,29,6}	{18,45,38,8}	{6,29,46,24}
Zielwavelet	{1,0}	←	←
AKF(<i>Inputwavelet</i>)	{3569,2612,972,144}	←	←
KKF(<i>Inp., Ziel</i>)	{24,0,0,0}	{18,0,0,0}	{6,0,0,0}

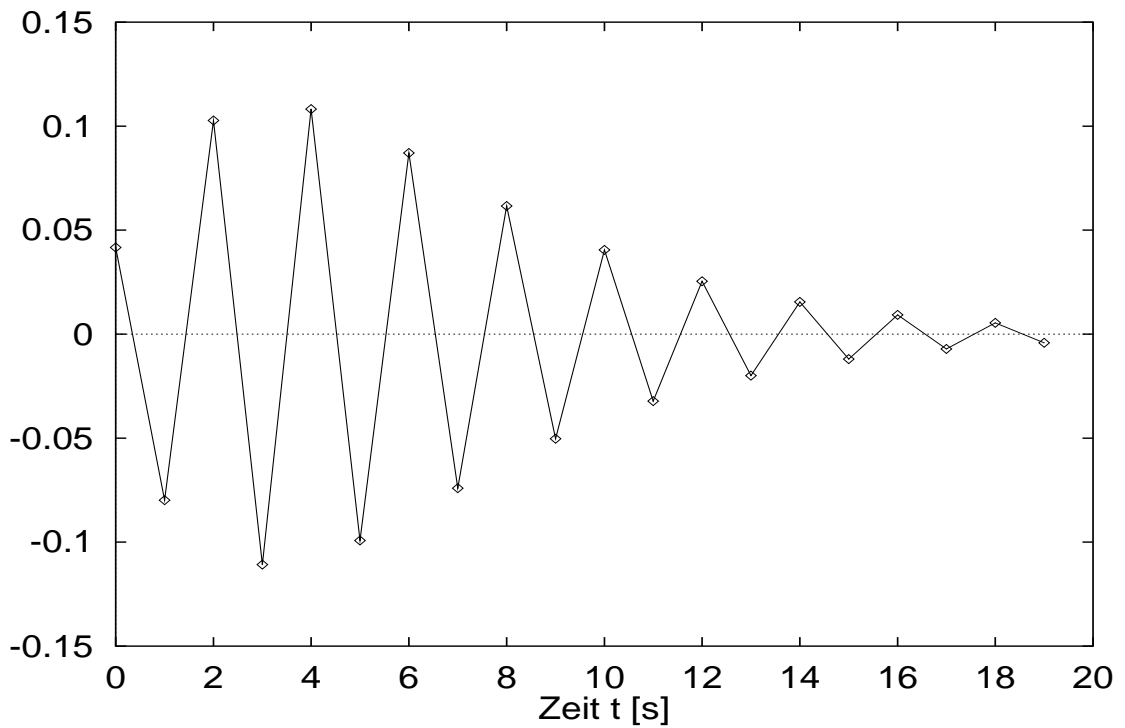
Inverse Filter:

Sample Nr.	minimalphasig	gemischtphasig	maximalphasig
0	0.026866589	0.0201499410	0.0067166472
1	-0.037718374	-0.0282887840	-0.0094295936
2	0.029206153	0.0219046180	0.0073015383
3	-0.012186326	-0.0091397464	-0.0030465815

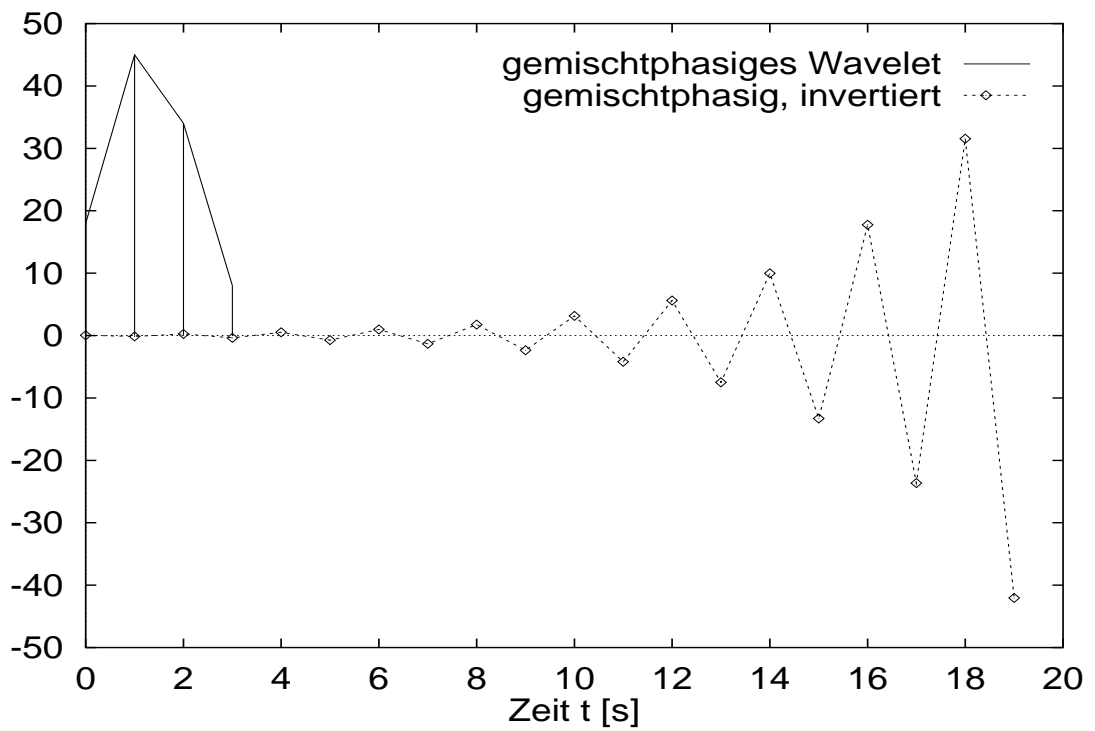
Faltung von Inputwavelet mit inversem Filter:

Sample Nr.	minimalphasig	gemischtphasig	maximalphasig
0	0.644798160	0.362698940	0.040299885
1	0.330622080	0.397549210	0.138205200
2	-0.254966500	-0.193614130	0.079316795
3	0.118378010	0.020573214	-0.079096645
4	0.060097158	0.107158140	0.021209657
5	-0.178166540	-0.135514440	0.035094157
6	-0.073117957	-0.073117971	-0.073117957
<hr/> <hr/> Fehlerenergie	<hr/> <hr/> 0.349856	<hr/> <hr/> 0.631955	<hr/> <hr/> 0.954354

13.4 Plots für Aufgabe 13:



Invertiertes minimalphasiges Wavelet



Invertiertes gemischtphasiges Wavelet

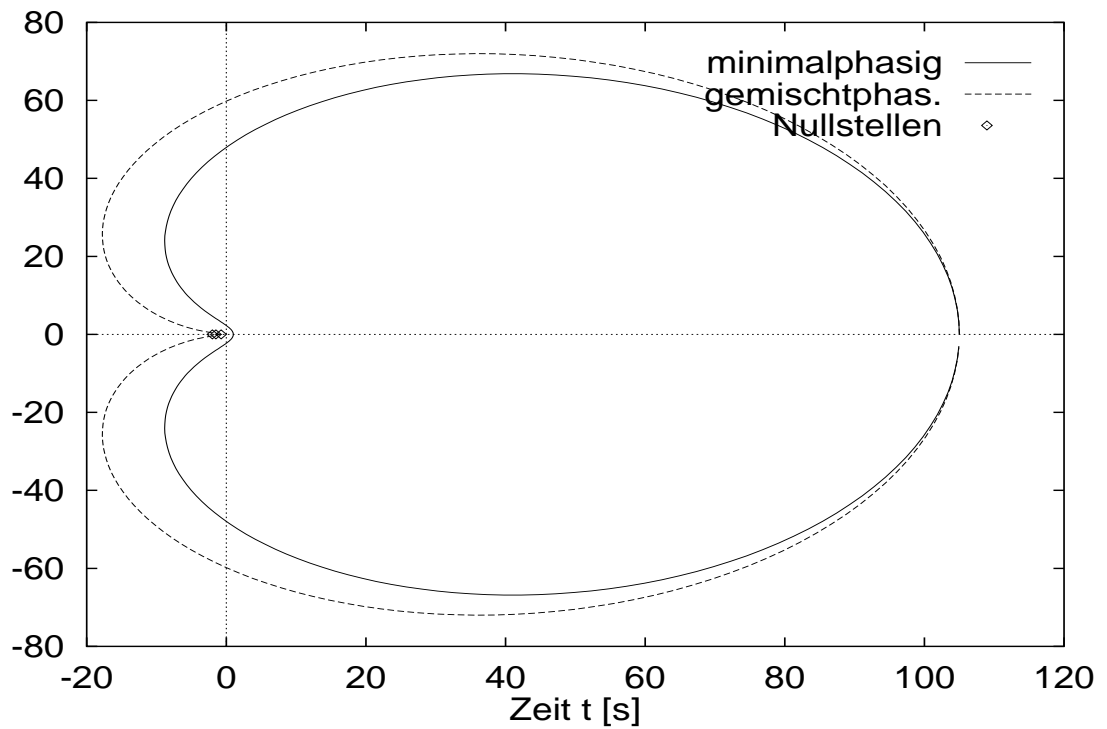


Abbildung des Einheitskreises durch Z-Transformierte

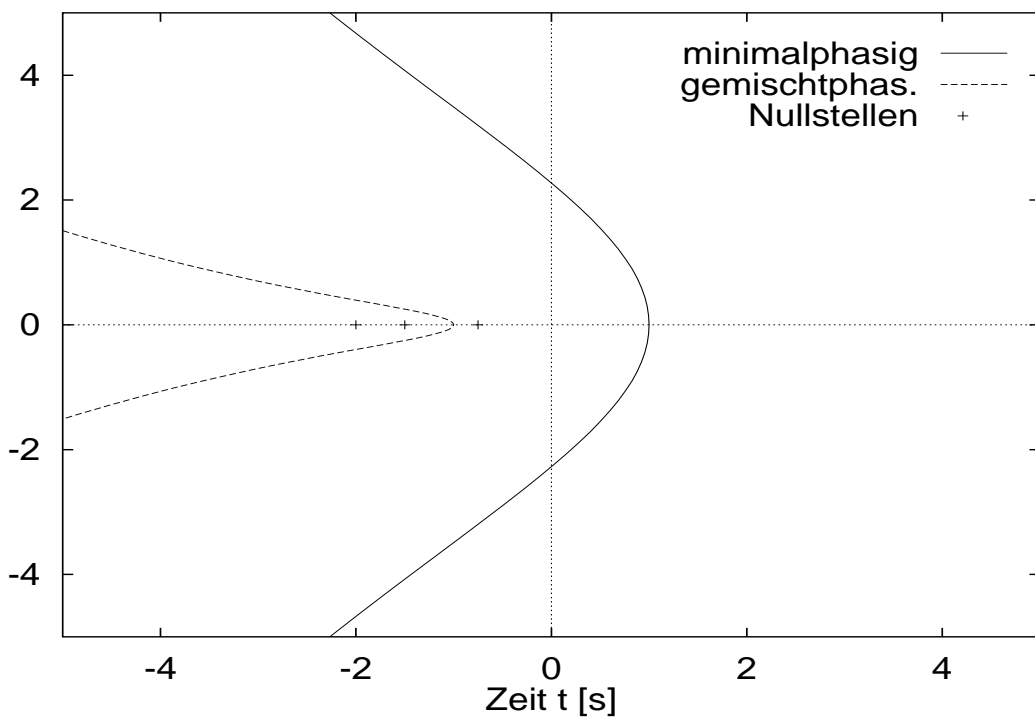


Abbildung des Einheitskreises durch Z-Transformierte
(Ausschnitt)

14 Wiener Optimum Filter und Spike- Dekonvolution

14.1 Aufgabenstellung:

- 1) Man schreibe ein FORTRAN / C Programm für das Wiener-Optimum-Filter unter Benutzung des Levinson-Algorithmus von S. 136 im Skript. Eingang: Wavelet X mit Länge LX . Gewünschter Ausgang: Wavelet Z mit Länge LZ . Berechnet werden sollen (mit Druckerplot-Ausgabe): AKF von X , KKF von X und Z für Zeiten ≥ 0 , Filteroperator F mit Länge $M = \max(LX, LZ)$ (F ist im allgemeinen ein Shaping-Filter), Faltung von X und F zum Vergleich mit Z . Man braucht funktionierende Programme für Korrelation und Faltung! Es ist empfehlenswert, X und Z mit Nullen bis zur Länge $MM = 2M$ zu verlängern.
- 2) Test des Programms mit dem Beispiel aus Abschnitt 10.1.2 im Skript. Achtung wegen unterschiedlicher Definitionen von n, p, m einerseits und LX, LZ und M andererseits.
- 3) Spike-Dekonvolution des minimal- und des maximalphasigen Wavelets aus Aufgabe 13, Vergleich der Fehlerenergien gemäß (10-1) im Skript.
- 4) Spike-Dekonvolution des Impulses

$$X(t) = \sin\left(\frac{2\pi}{T_0} \cdot t \cdot \exp -(t/T_1)^2\right)$$

$$T_0 = 20\Delta t, \quad T_1 = 15\Delta t, \quad LX = 30$$

Der Zeitindex 1 muß der Zeit $t = \Delta t$ entsprechen, d.h. $X(1)NICHT = 0$. Man versuche auch die Spike-Dekonvolution des umgedrehten Impulses, sowohl mit $Z(t) = \delta(t)$ als auch mit $Z(t) = \delta(t - 24\Delta t)$ bzw. $Z(t) = \delta(t - 34\Delta t)$.

14.2 Bemerkungen zum Levinson-Algorithmus

Levinson Algorithmus - Implementierung in C

Im Anhang steht der C Quelltext einer **1 zu 1** Übersetzung der Fortran-Subroutine **LEVIN** von Seite 136 im Skript.

Ein Algorithmus zur Lösung von Gleichungssystemen mit den allgemeineren Toeplitzmatrizen findet man z.B. in den Numerical Recipies, File **toeplz.c** .

Parameter in *toeplz()*

```
void toeplz(float *r,float *x,float *y,int n)
```

r : Feld von 'floats' für die Töpliz-Matrix???

x : Feld von 'floats' für zu berechnenden Vektor???

y : Feld von 'floats' für den bekannten Vektor???

n : Anzahl der Vektor-Elemente ???

14.3 Test mit dem Beispiel aus Abschnitt 10.1.2

Eingang des Filters ist: (1; 0.5)

Gewünschter Ausgang ist: (1; 0) (Deltaimpuls)

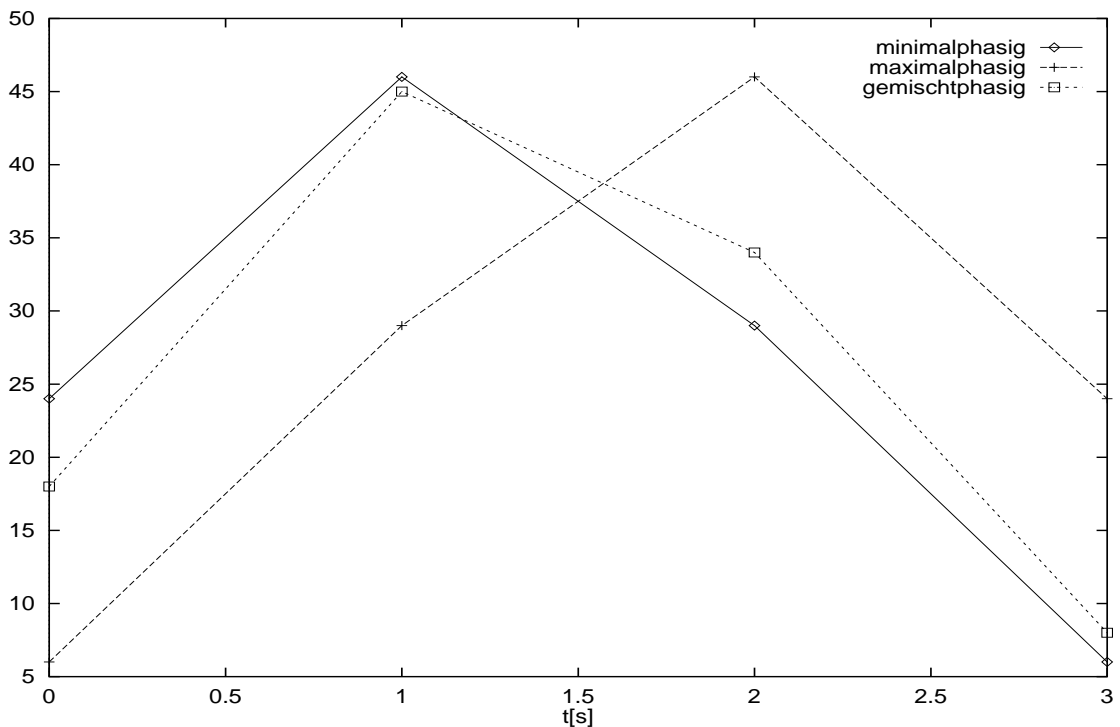
AKF des Eingangs ist: $\Phi_0 = 1.25$; $\Phi_1 = 0.5$

KKF von Eingang und gewünschtem Ausgang ist: $\phi_0 = 1$; $\phi_1 = 0$

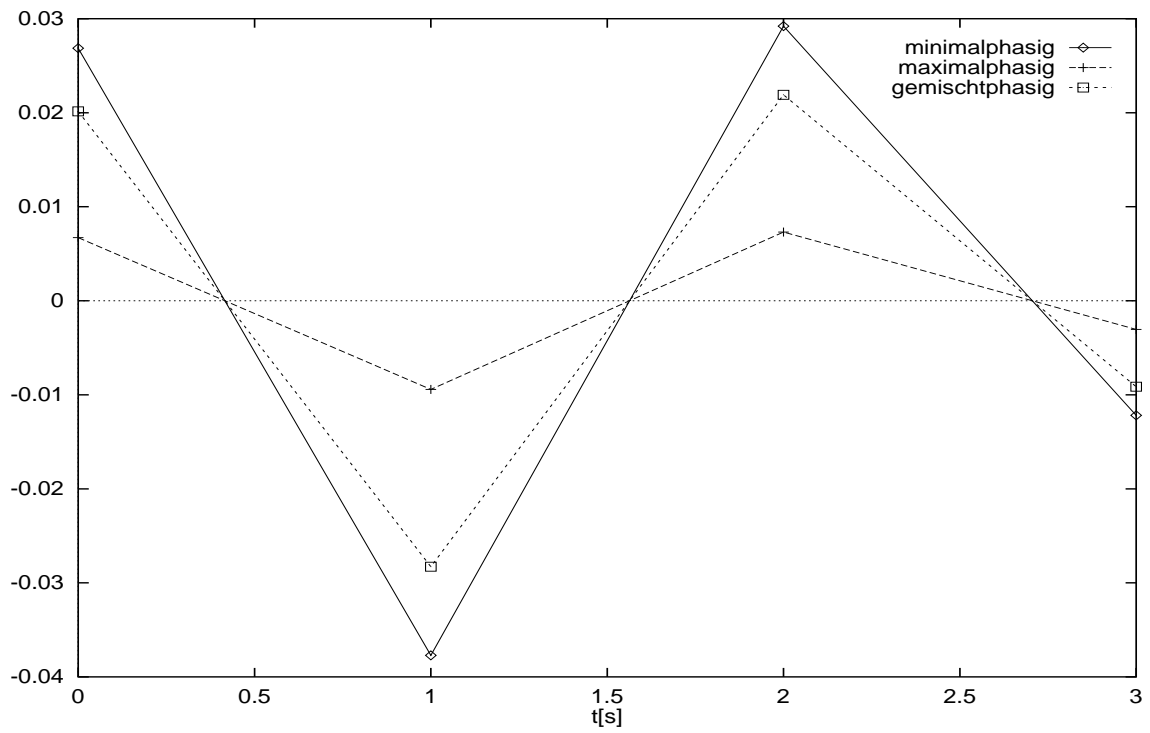
Die Lösung des Gleichungssystems liefert die Filterkoeffizienten:

$f_0 = 0.952$; $f_1 = -0.381$

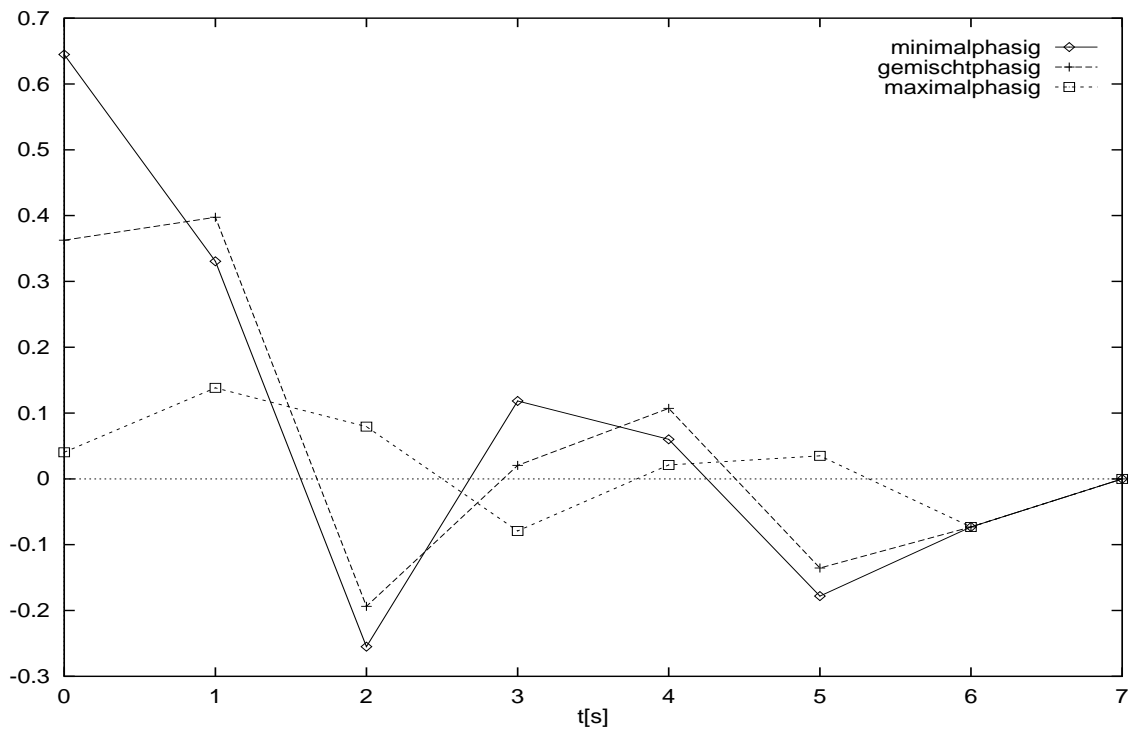
14.4 Plots für Aufgabe 14:



minimal- gemischt- und maximalphasiges Wavelet aus Aufgabe 13

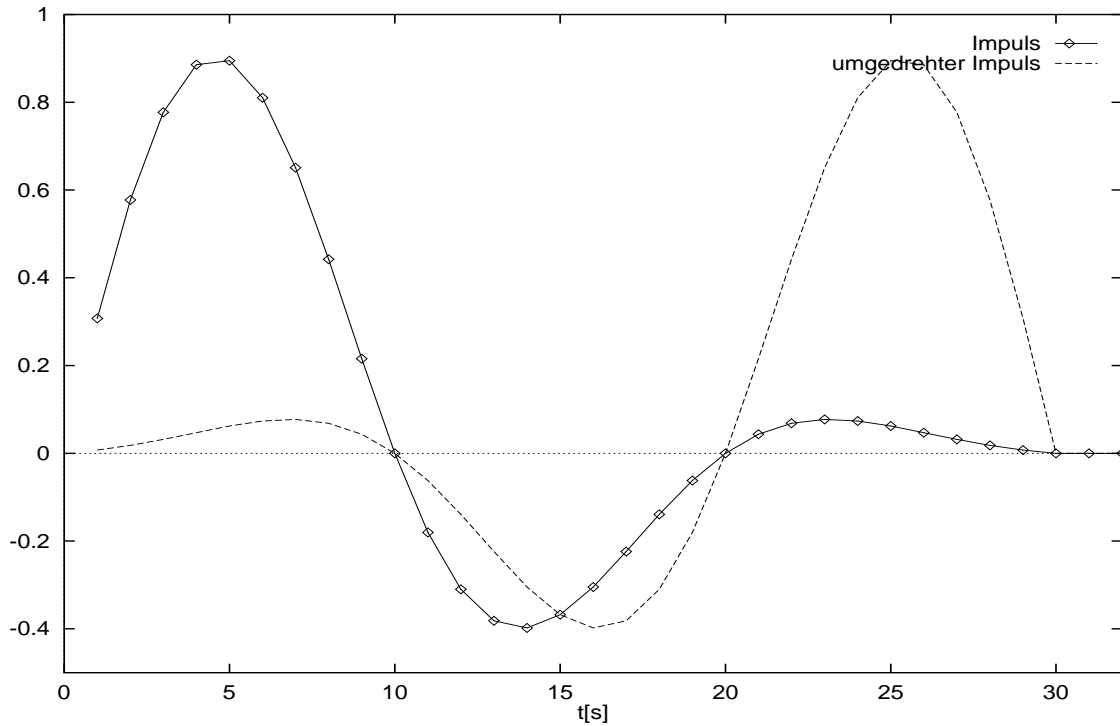


Spike-Dekonvolution des minimal- gemischt- und maximalphasigen Wavelets

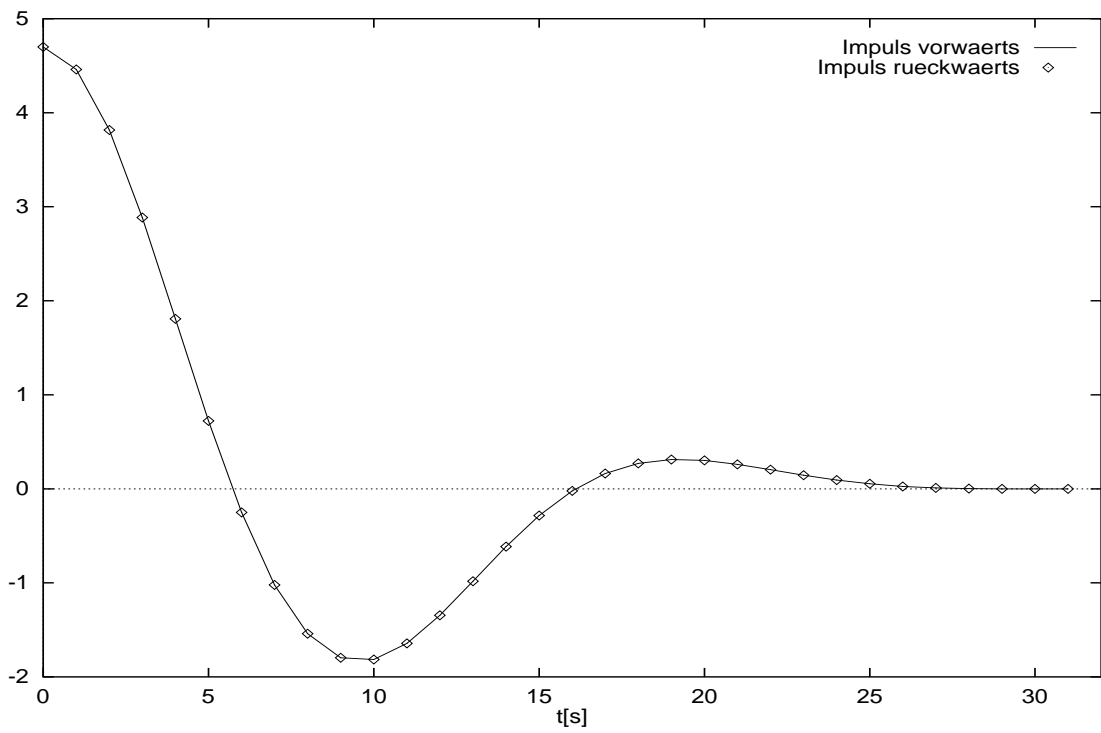


Faltung von inversem Filter mit Eingangswavelet

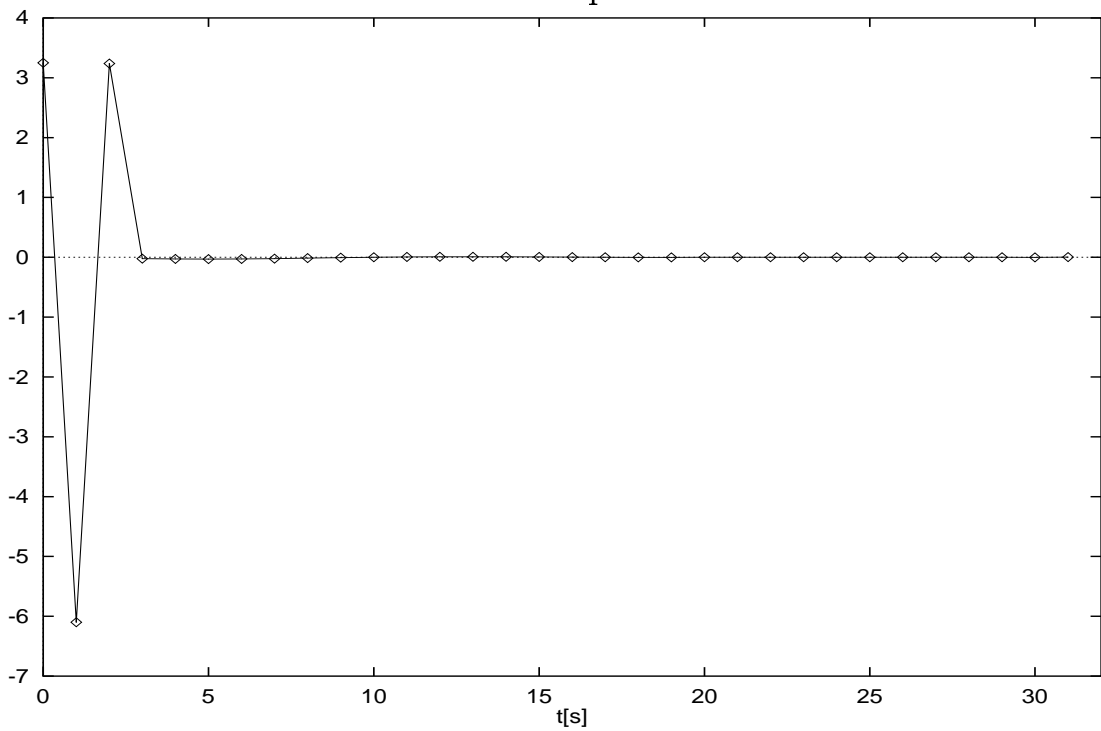
Fehlerenergie:	minphas	mixphas	maxphas
	0.350	0.637	0.960



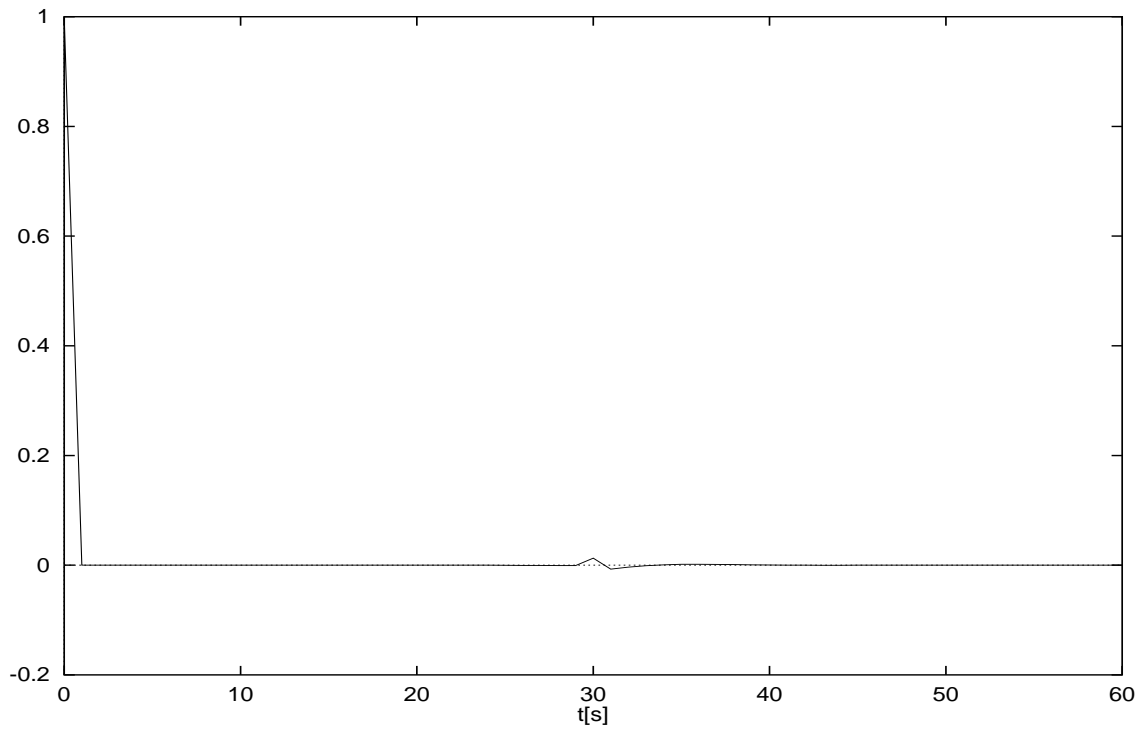
Impuls $\sin\left(\frac{2\pi}{T_0} \cdot t\right) \cdot \exp\left(-\left(t/T_1\right)^2\right)$ vorwärts und rückwärts



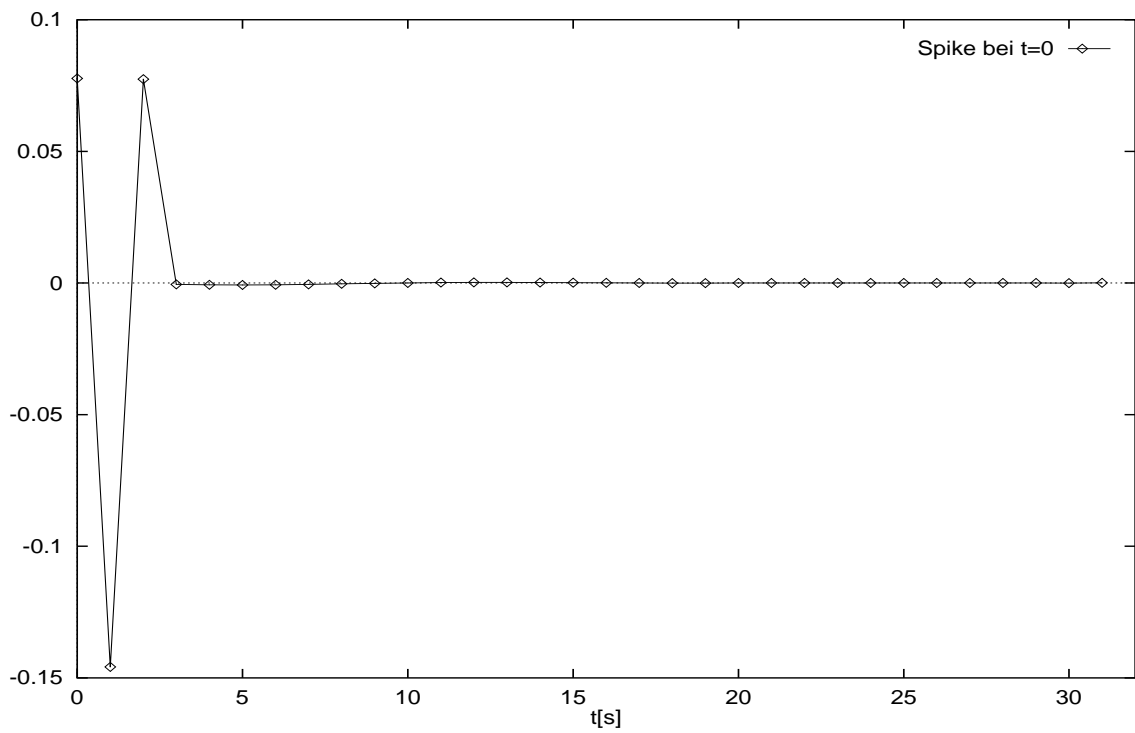
AKF des Impulses



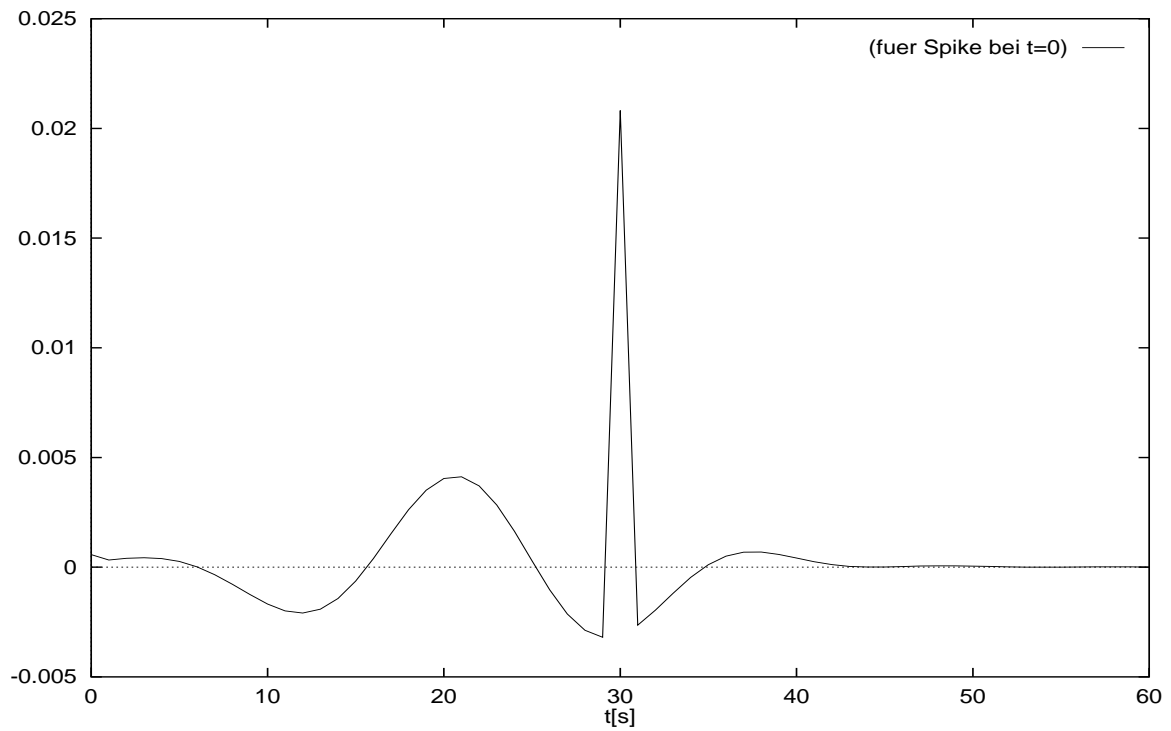
Inverses Filter des Impulses



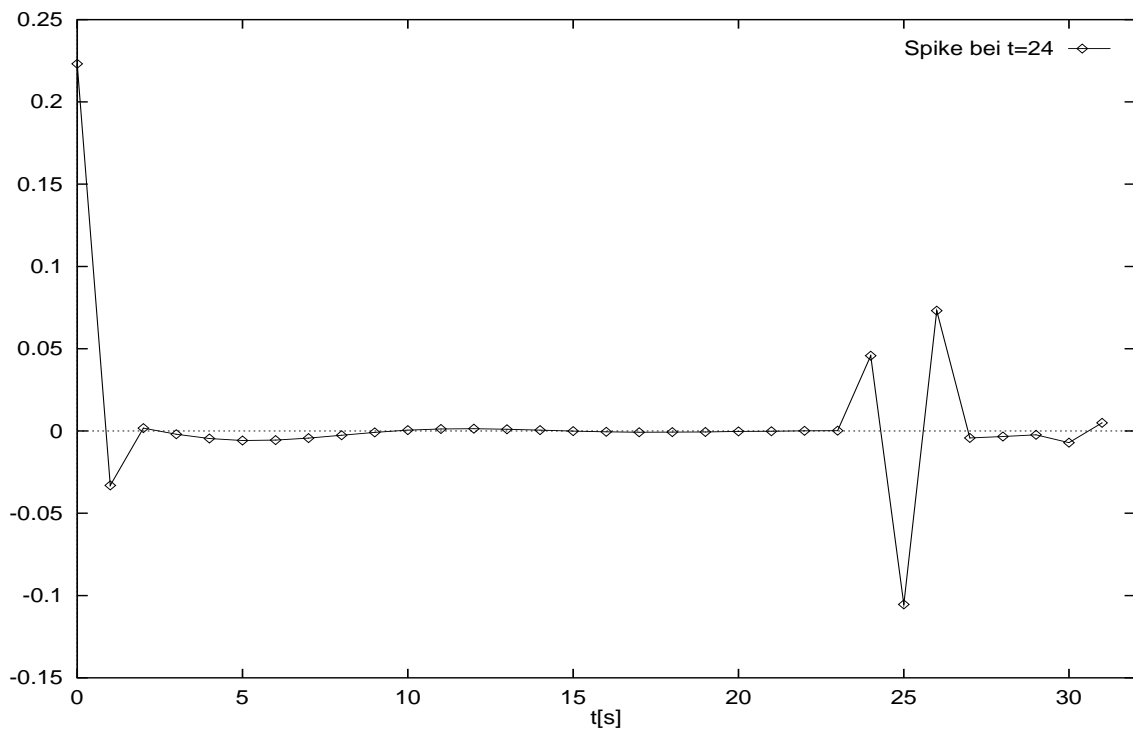
Spike-Dekonvolvierte des Impulses
Fehlerenergie: 0.000219



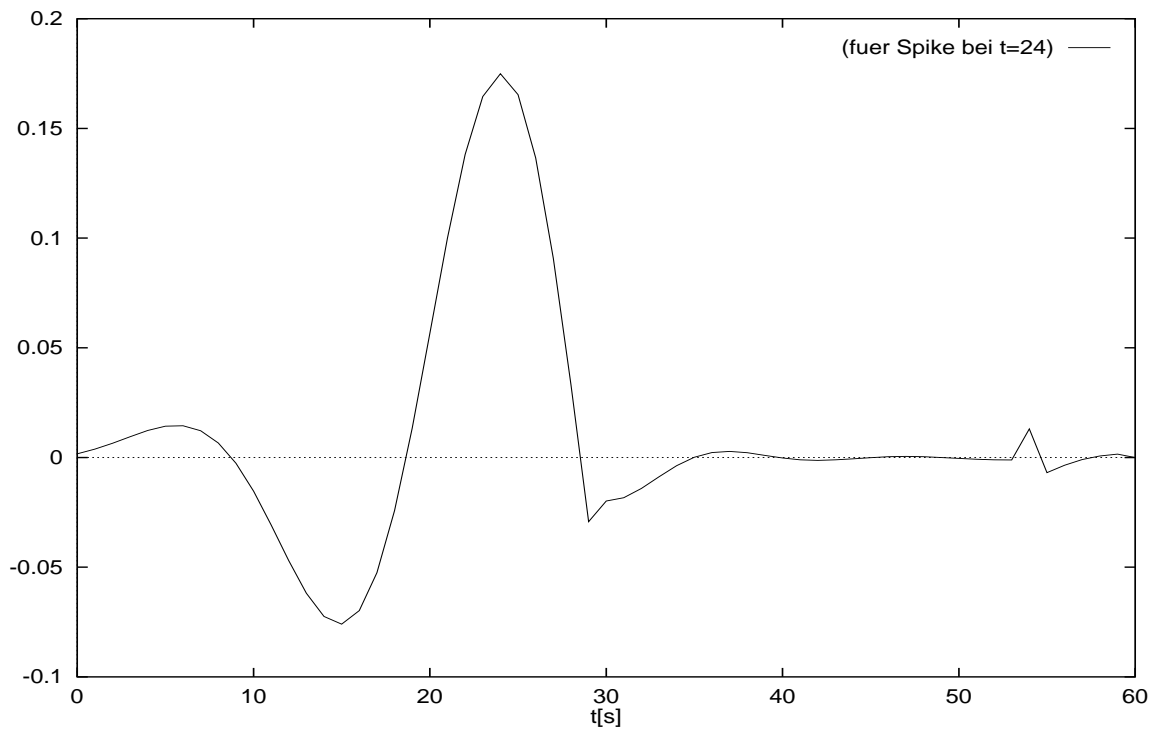
Inverses Filter des umgedrehten Impulses



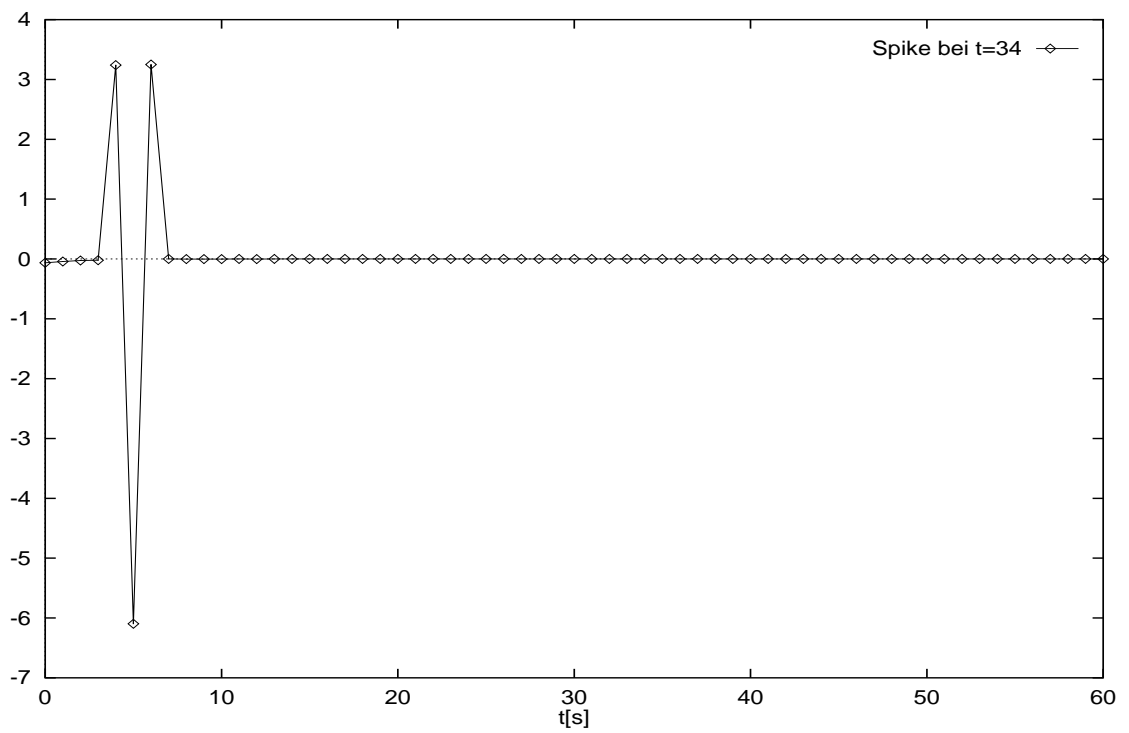
Spike-Dekonvolvierte des umgedrehten Impulses
Fehlerenergie: 0.999 (Spike bei $t = 0s$)



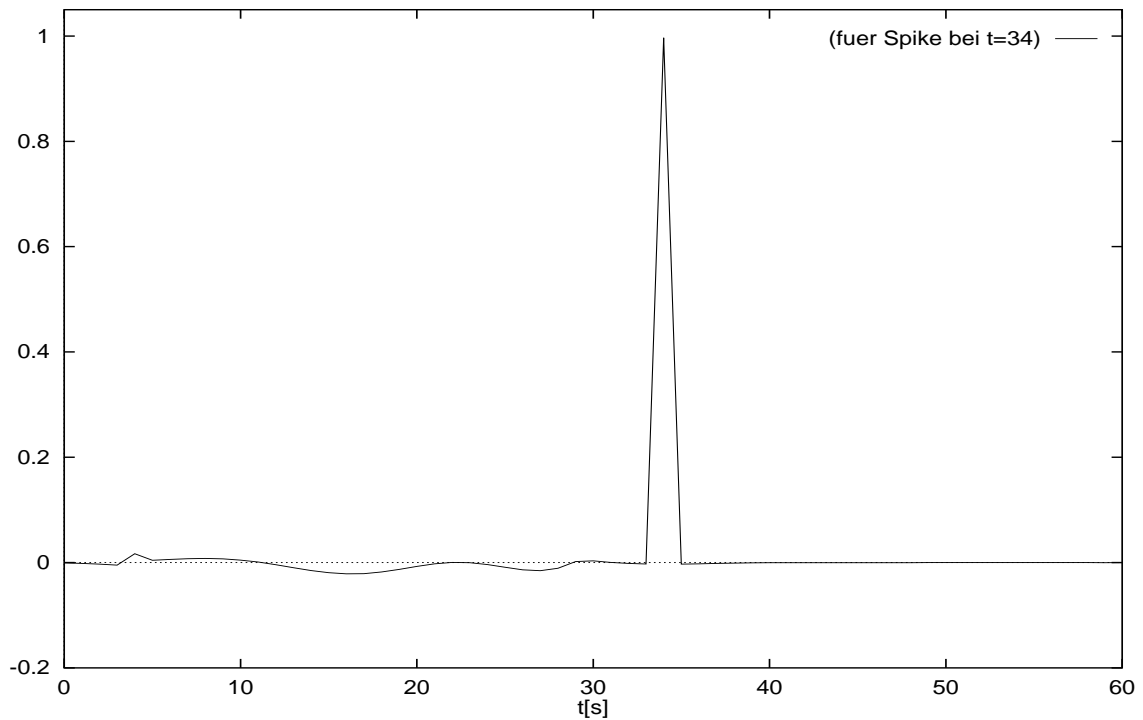
Inverses Filter des umgedrehten Impulses, Spike bei $t = 24s$



Spike-Dekonvolvierte des umgedrehten Impulses
 Fehlerenergie: 0.825 (Spike bei $t = 24s$)



Inverses Filter des umgedrehten Impulses, Spike bei $t = 34s$



Spike-Dekonvolvierte des umgedrehten Impulses
 Fehlerenergie: 0.003410 (Spike bei $t = 34s$)

15 Wiener Filter zur Unterdrückung von Rauschen

15.1 Aufgabenstellung:

Man schreibe ein FORTRAN / C Programm zur Unterdrückung von Rauschen mit einem Wiener-Filter $f(t)$ und wende es auf folgenden Fall an:

Signal sei eine Sinusschwingung (eine Periode), die nach $50 \cdot \Delta t$ einsetzt und $20 \cdot \Delta t$ lang ist, also bis $70 \cdot \Delta t$ geht. Das Signal soll insgesamt $120 \cdot \Delta t$ lang sein. (Siehe Skizze im Skript)

Rauschen $n(t) =$ Pseudo-Zufallszahlen (vgl. Aufgabe 9) mit Werten zwischen

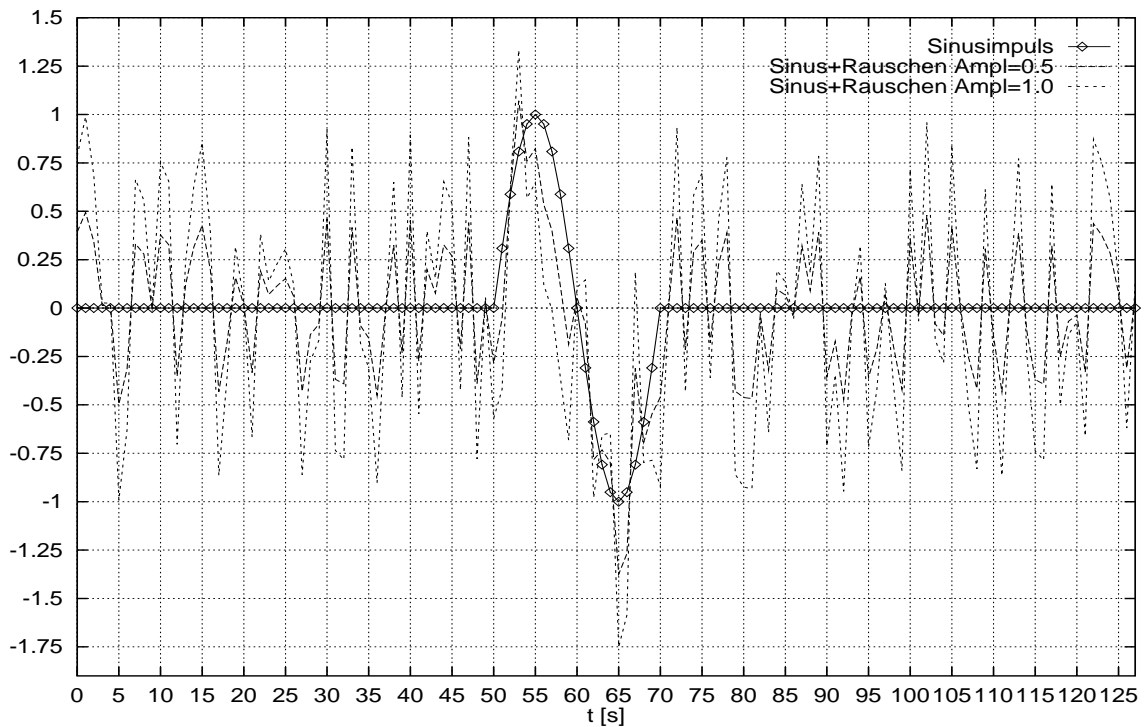
- a) -0.5 und $+0.5$,
- b) -1.0 und $+1.0$ (Verdopplung)

Man erzeuge Druckerplots der Zeitreihe $x(t) = s(t) + n(t)$, der AKFs $\Phi_{nn}(t)$ und $\Phi(t)$, der KKF $\varphi(t)$ und des Filters $f(t)$. Dann filtere man $x(t)$ durch Faltung mit $f(t)$. Bei Φ_{nn}, Φ, φ und f reichen 50 Punkte.

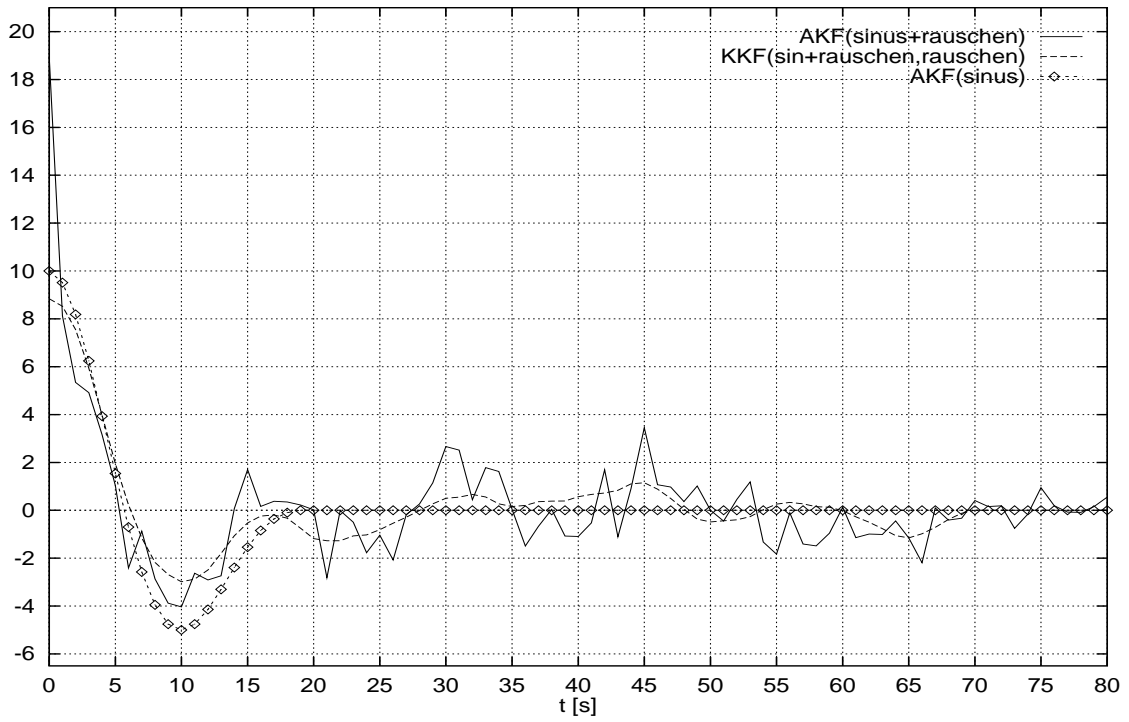
Man untersuche folgende Spezialfälle:

- 1) $\Phi = \Phi_{ss} + \Phi_{nn}$, $\varphi = \Phi_{ss}$ (idealer Fall: s und n unkorreliert)
- 2) $\Phi = \Phi_{xx}$, $\varphi = \Phi_{xx} - \Phi_{nn}$ (realer Fall)

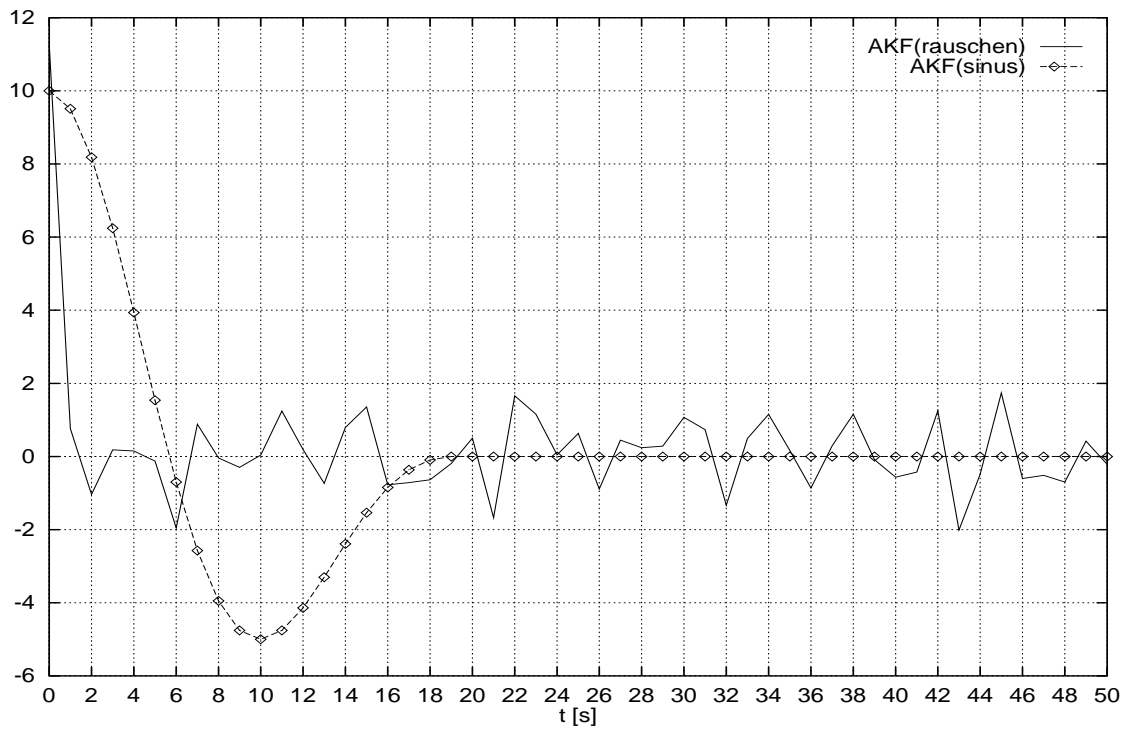
15.2 Plots für Aufgabe 15:



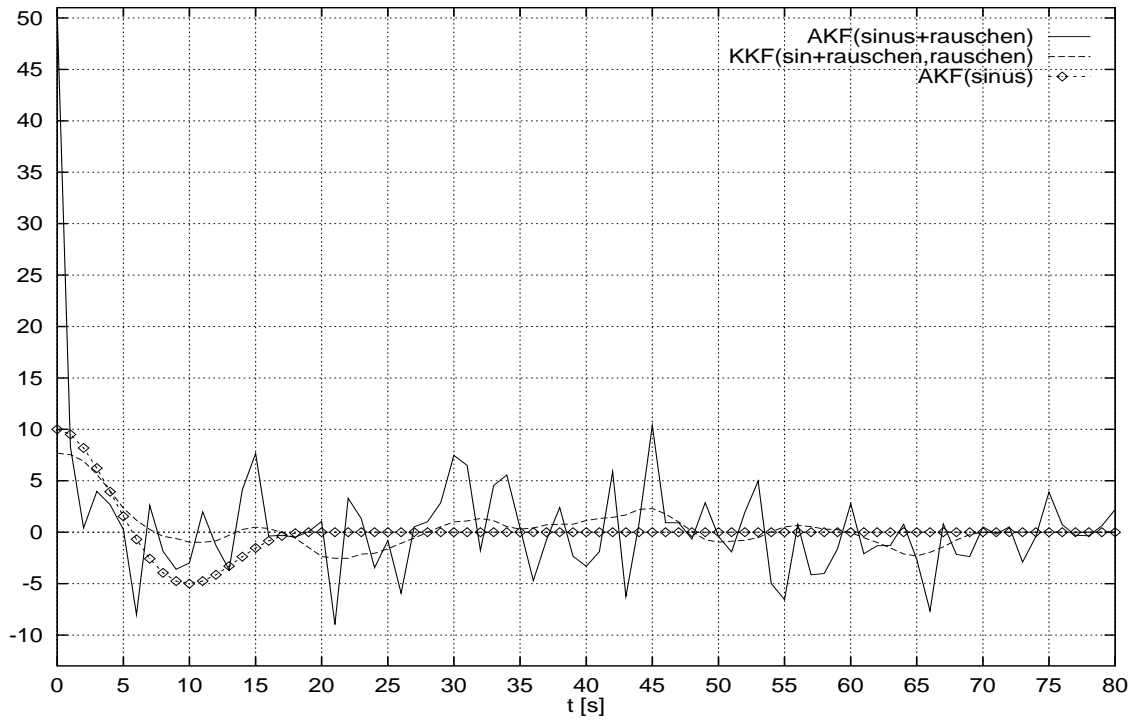
Sinusimpuls überlagert von Rauschen



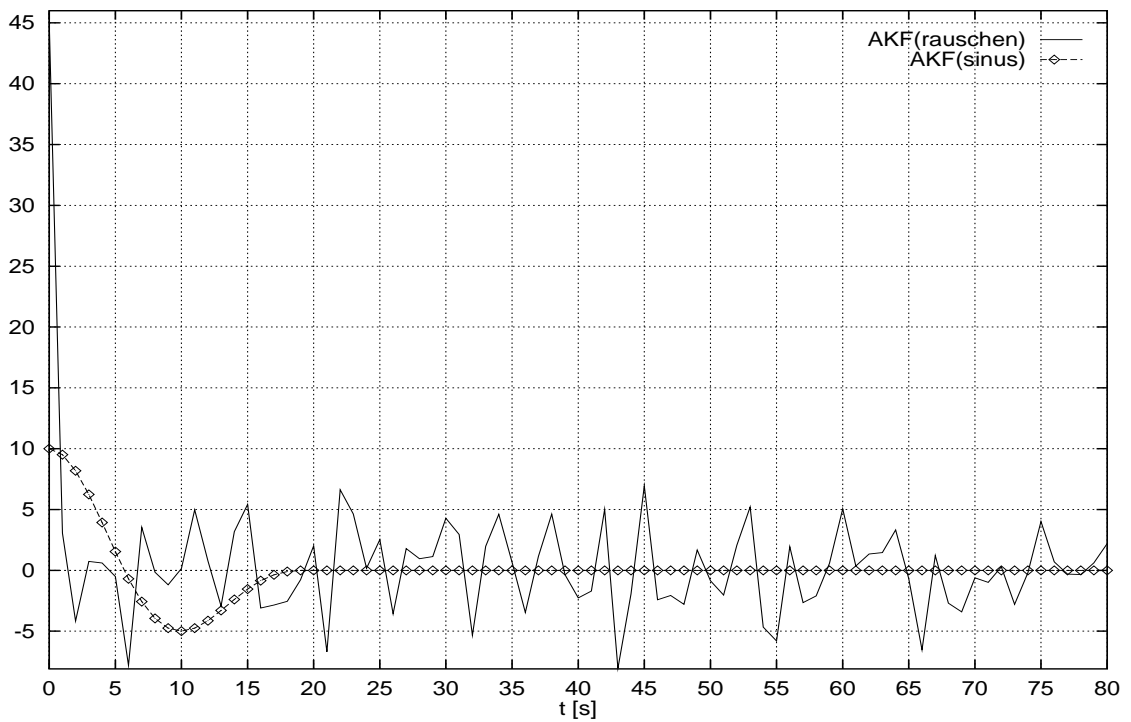
Korrelationsfunktionen fuer Rauschamplitude 0.5



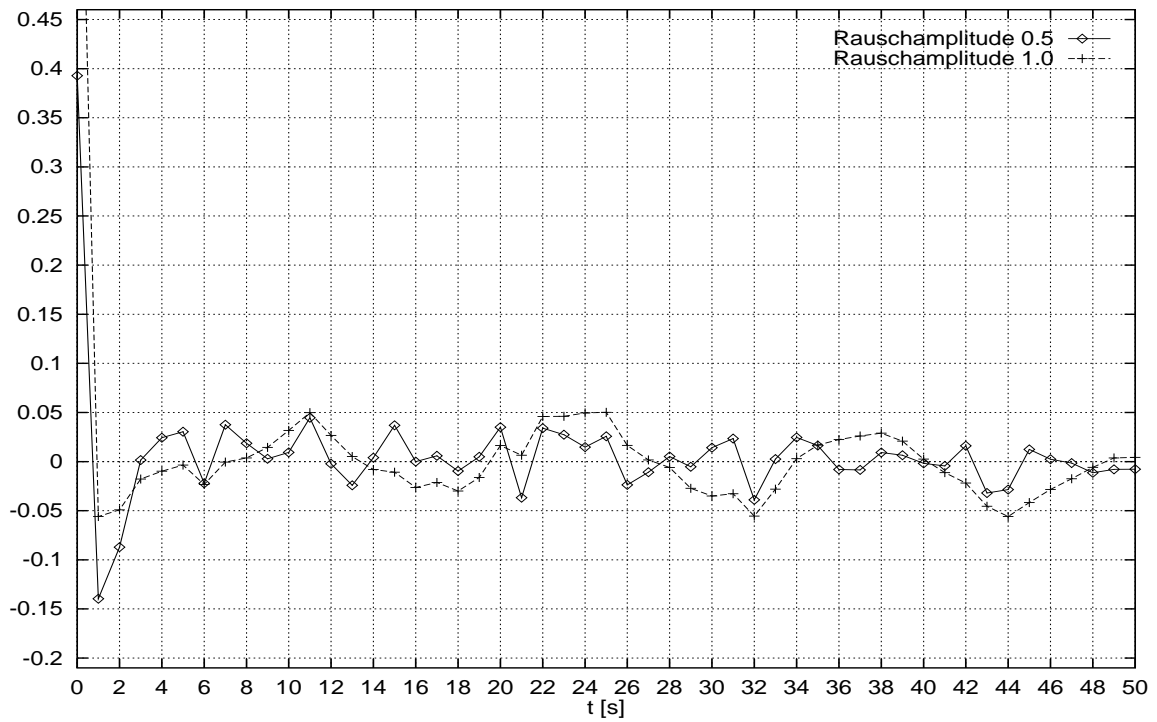
Korrelationsfunktionen fuer Rauschamplitude 0.5



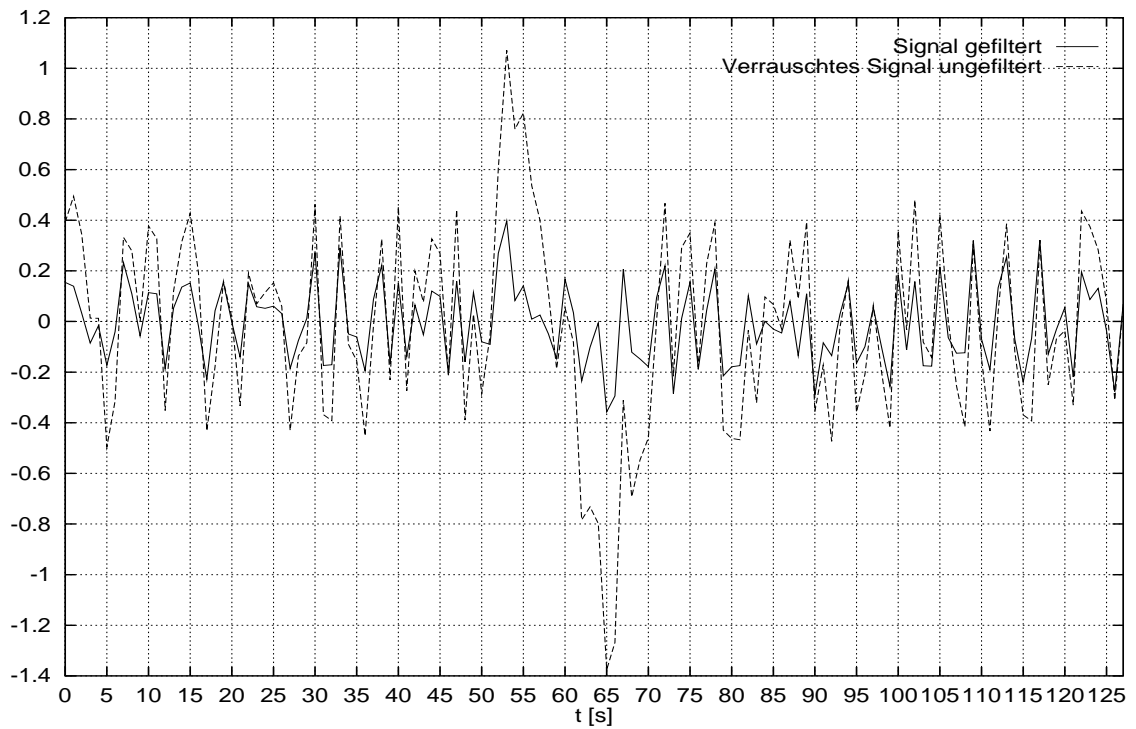
Korrelationsfunktionen fuer Rauschamplitude 1.0



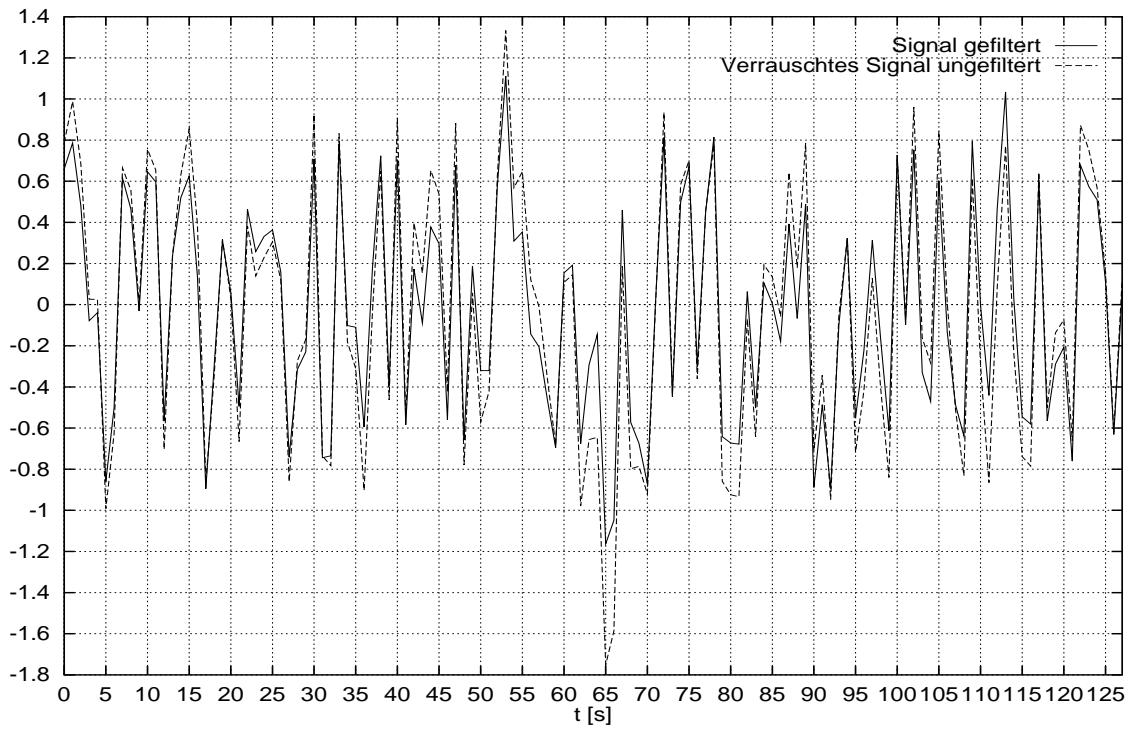
Korrelationsfunktionen fuer Rauschamplitude 1.0



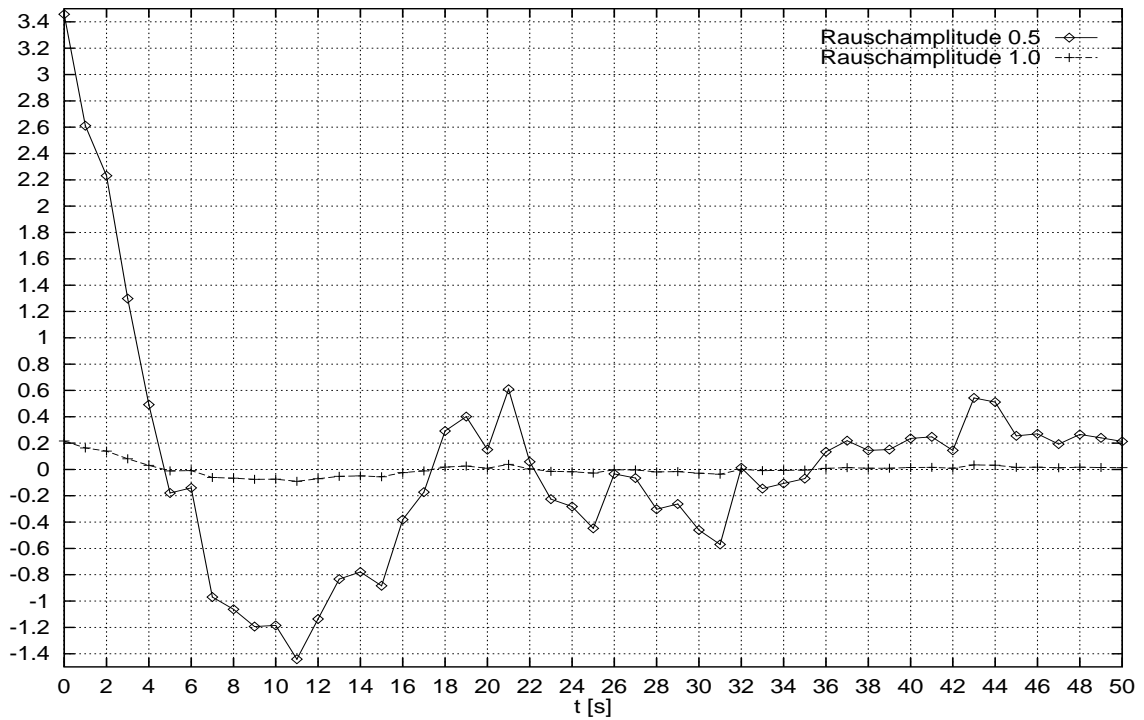
Realer Fall: Filteroperator



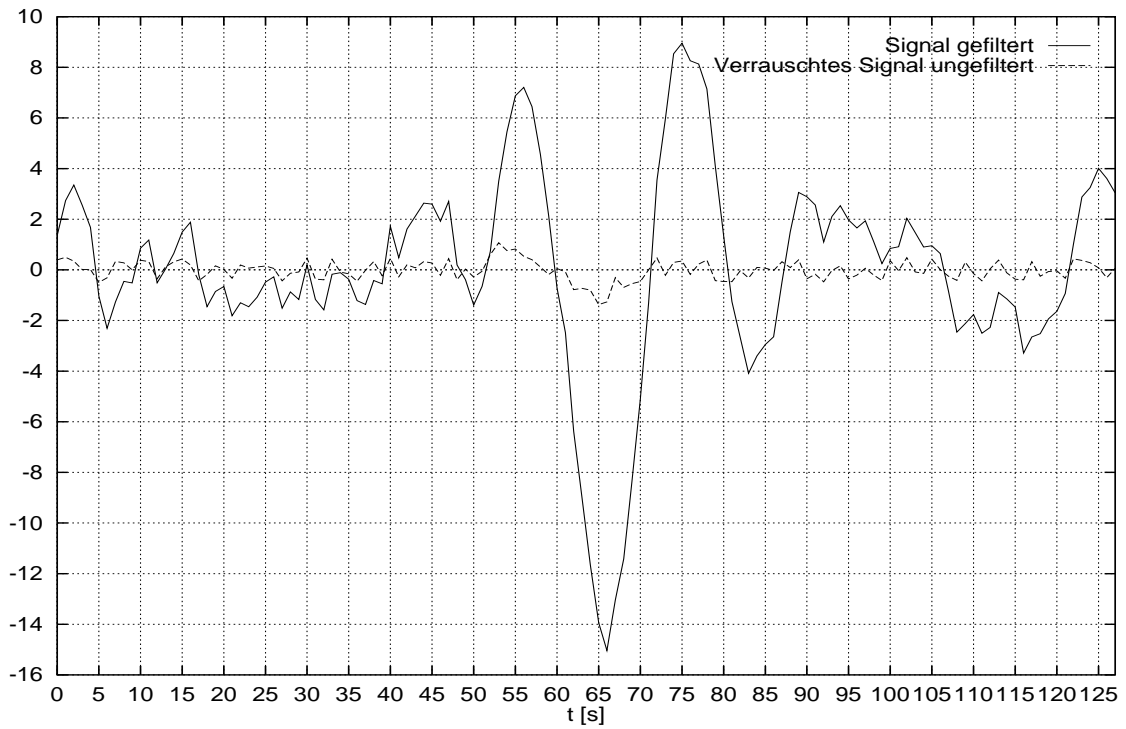
Realer Fall: Faltung von Signal mit Filteroperator - Rauschamplitude 0.5



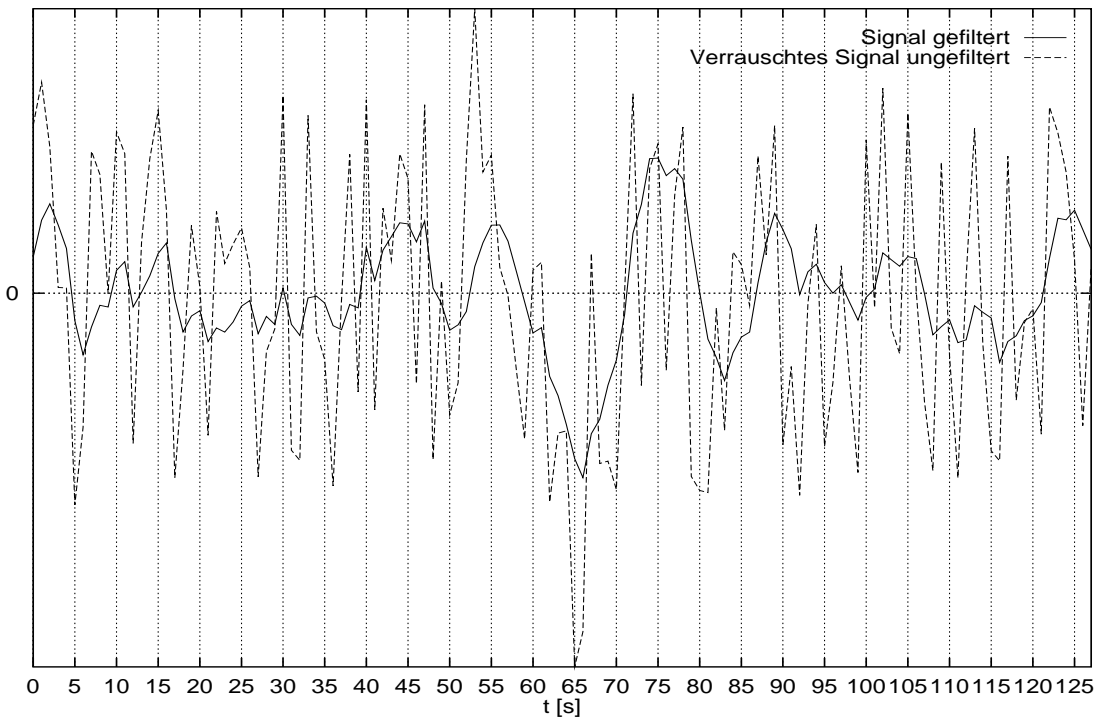
Realer Fall: Faltung von Signal mit Filteroperator - Rauschamplitude 1.0



Idealer Fall: Filteroperator



Idealer Fall: Faltung von Signal mit Filteroperator - Rauschamplitude 0.5



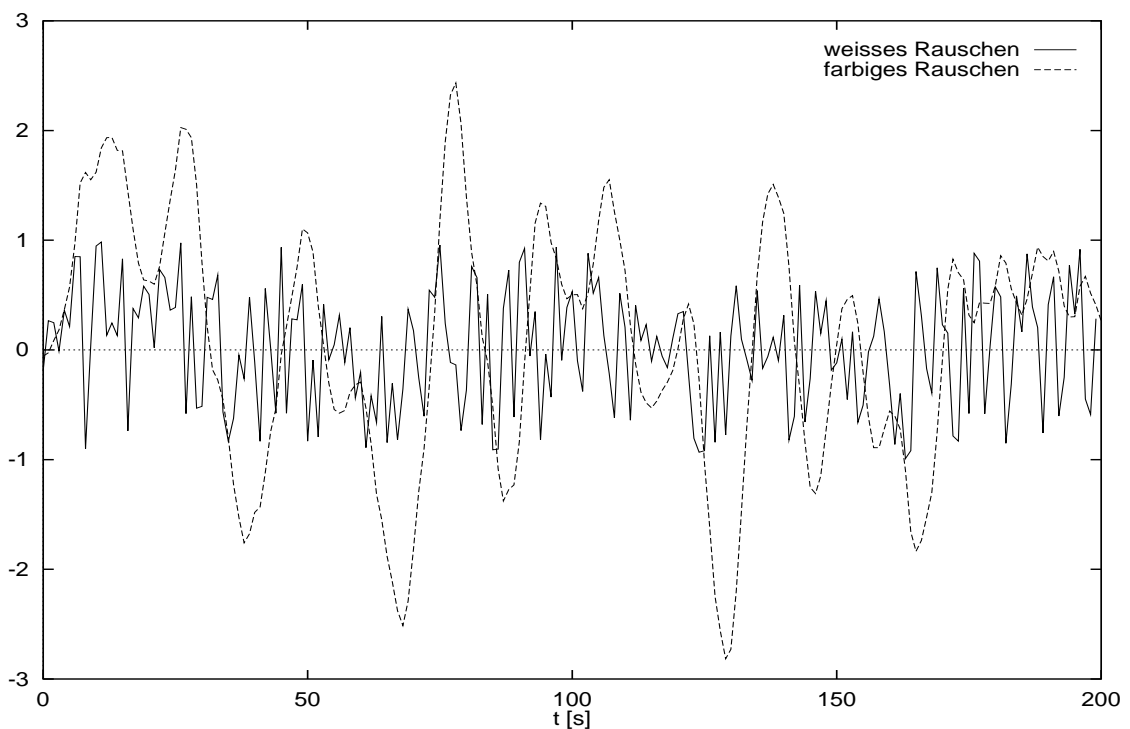
Idealer Fall: Faltung von Signal mit Filteroperator - Rauschamplitude 1.0

16 Farbiges Rauschen / prädikative Dekonvolution

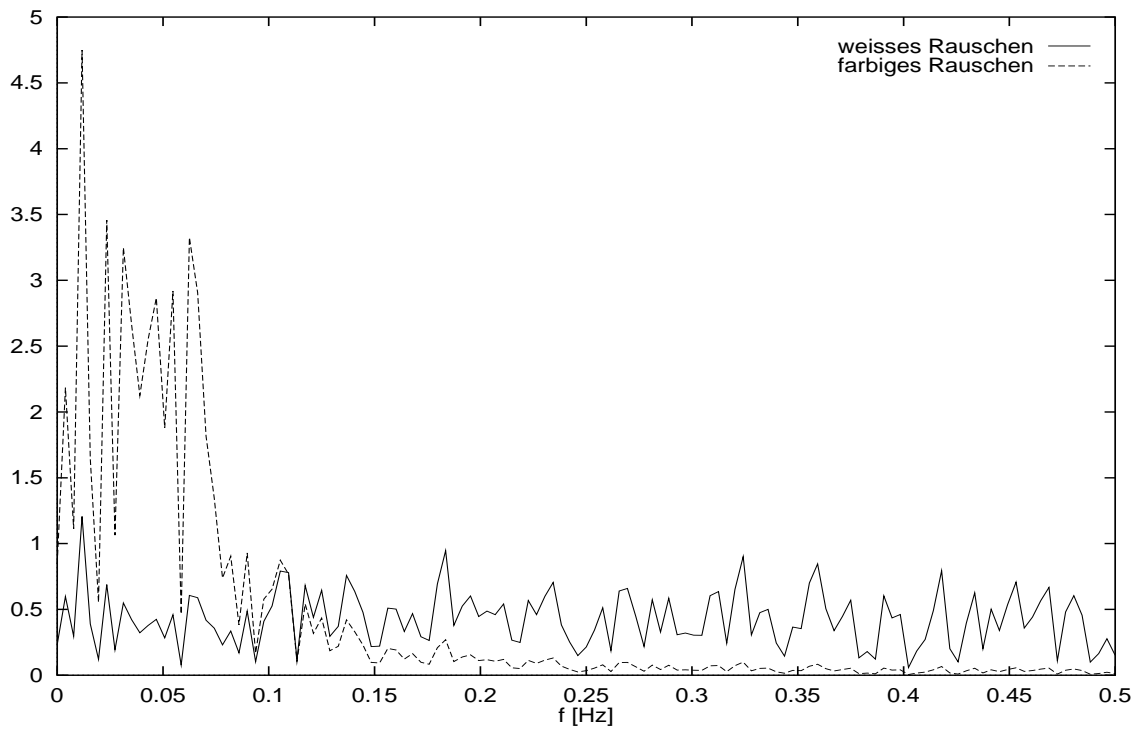
16.1 Aufgabenstellung:

- 1) Man erzeuge eine Zeitreihe, die genähert farbiges Rauschen repräsentiert, indem man eine Folge von 200 Pseudo-Zufallszahlen mit dem Wavelet aus Aufgabe 14, Teil 4) faltet.
- 2) Man berechne das Vorhersage-Filter $a(t)$ und das Vorhersagefehler-Filter $b(t)$ für die Vorhersagedistanz $\alpha = 1$. Die AKF sei die des Wavelets.
- 3) Man mache eine Vorhersagefehler-Filterung des farbigen Rauschens mit $b(t)$ und vergleiche mit der Folge von Zufallszahlen.
- 4) Wiederholung von 2) und 3) mit der passend abgeschnittenen AKF des farbigen Rauschens ($\alpha = 1$ auch in diesem Fall).

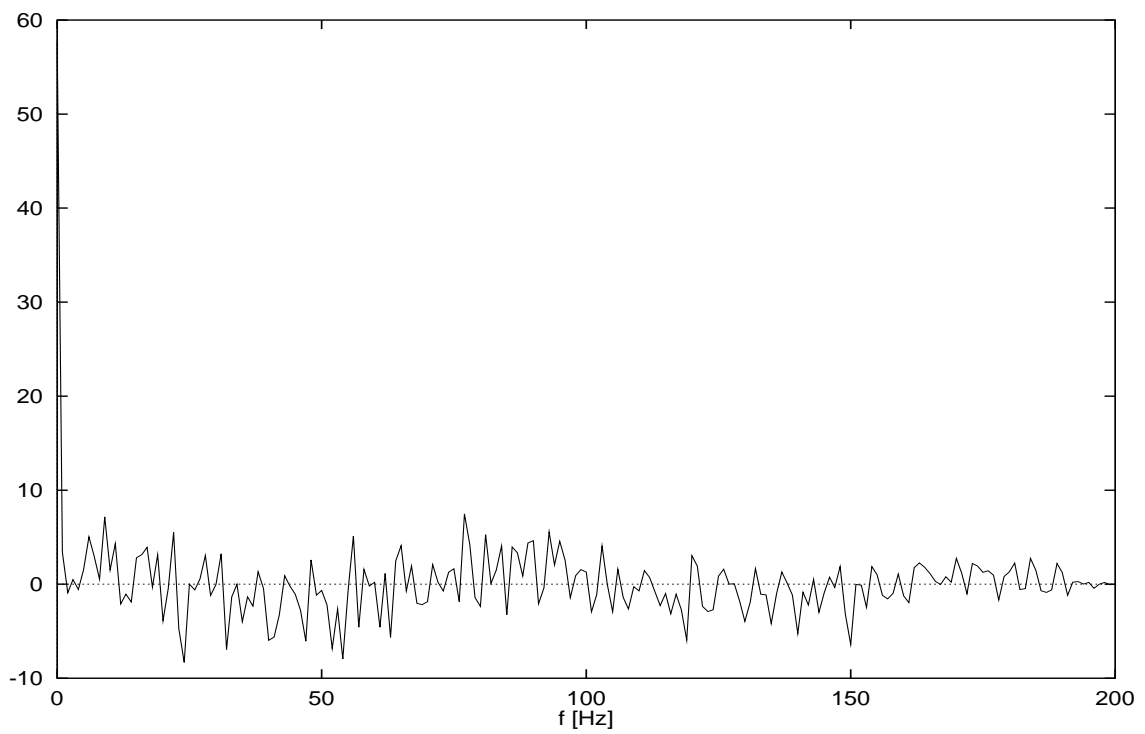
16.2 Plots für Aufgabe 16:



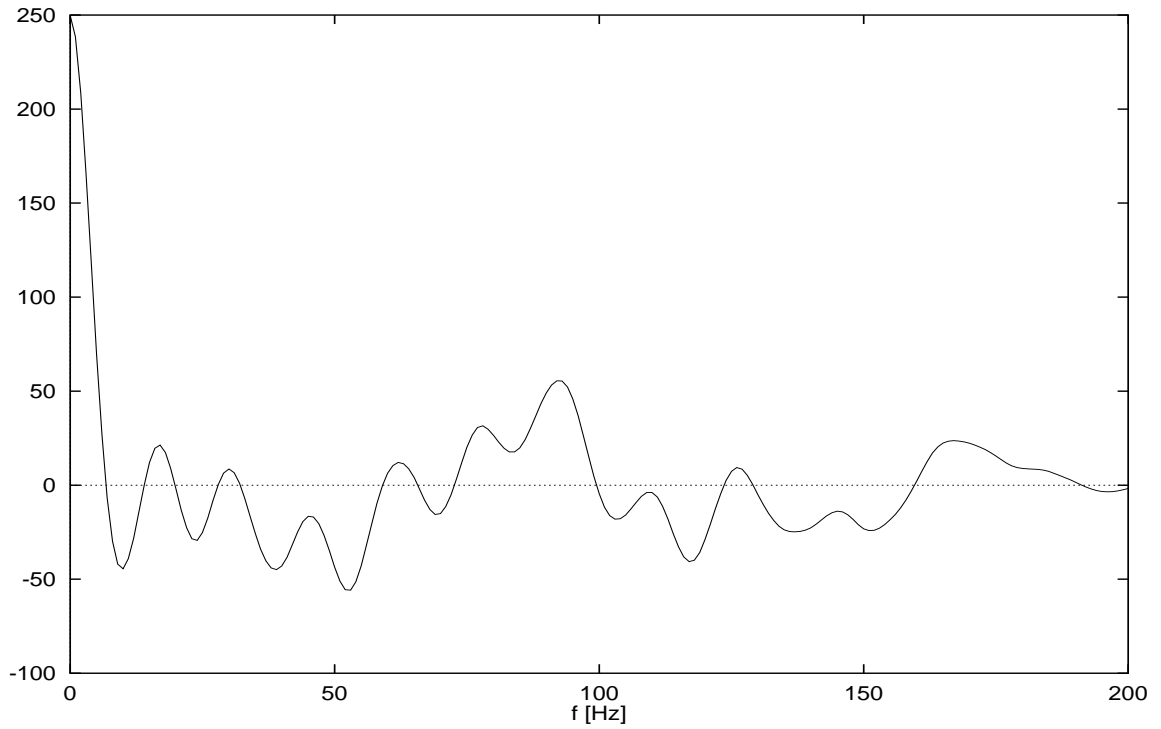
Rauschen im Zeitbereich



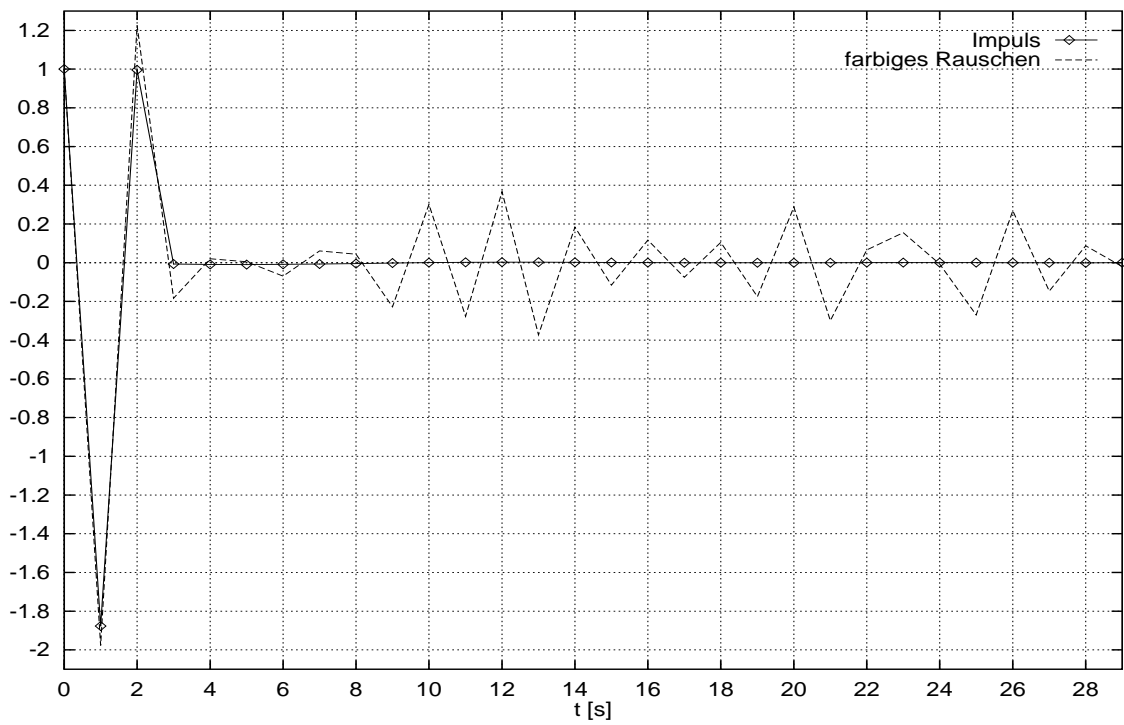
Amplitudenspektren von weißem und farbigem Rauschen



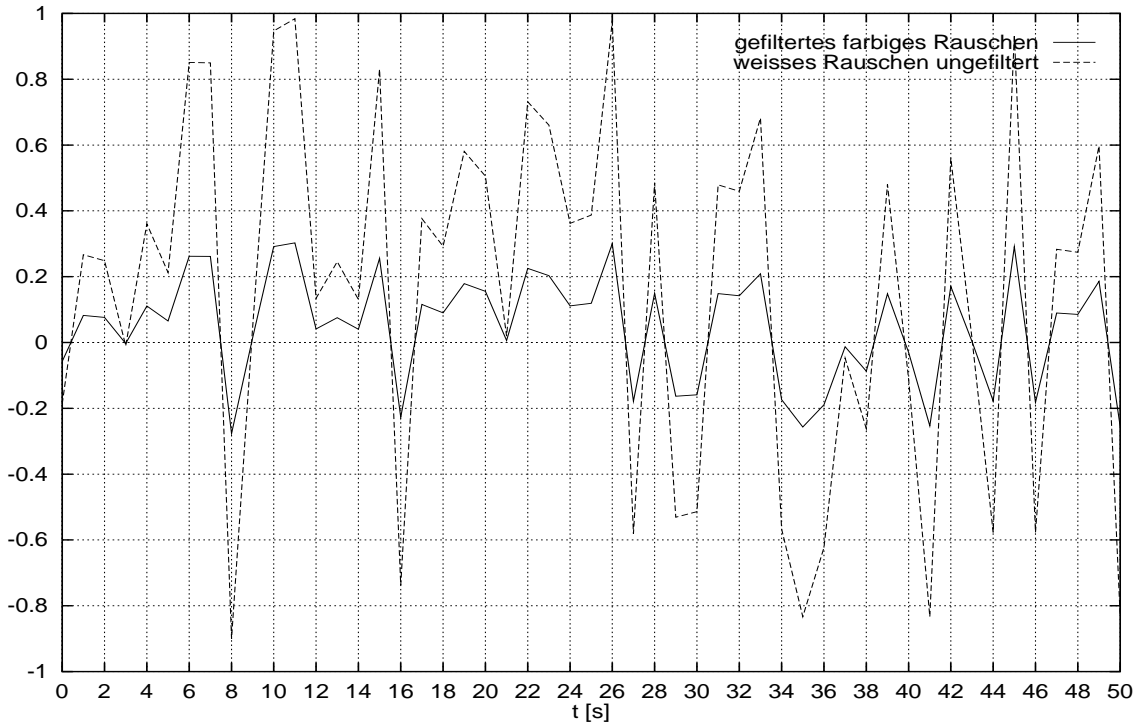
AKF des weißen Rauschens



AKF des farbigen Rauschens



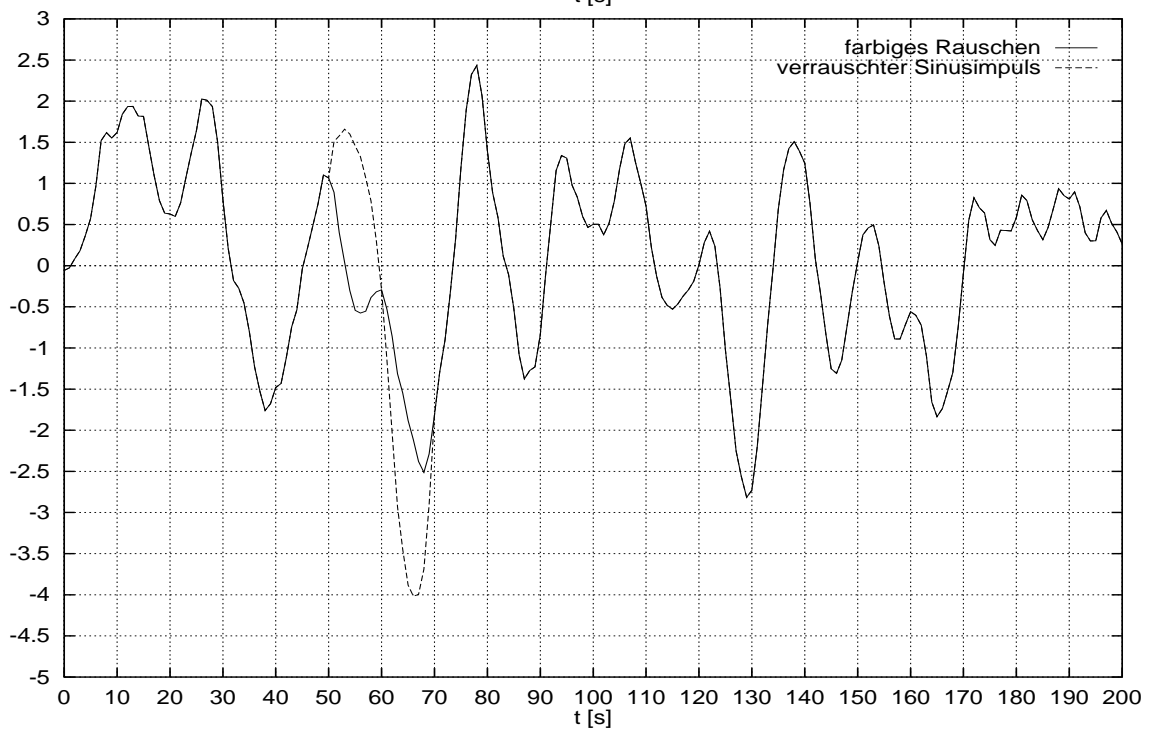
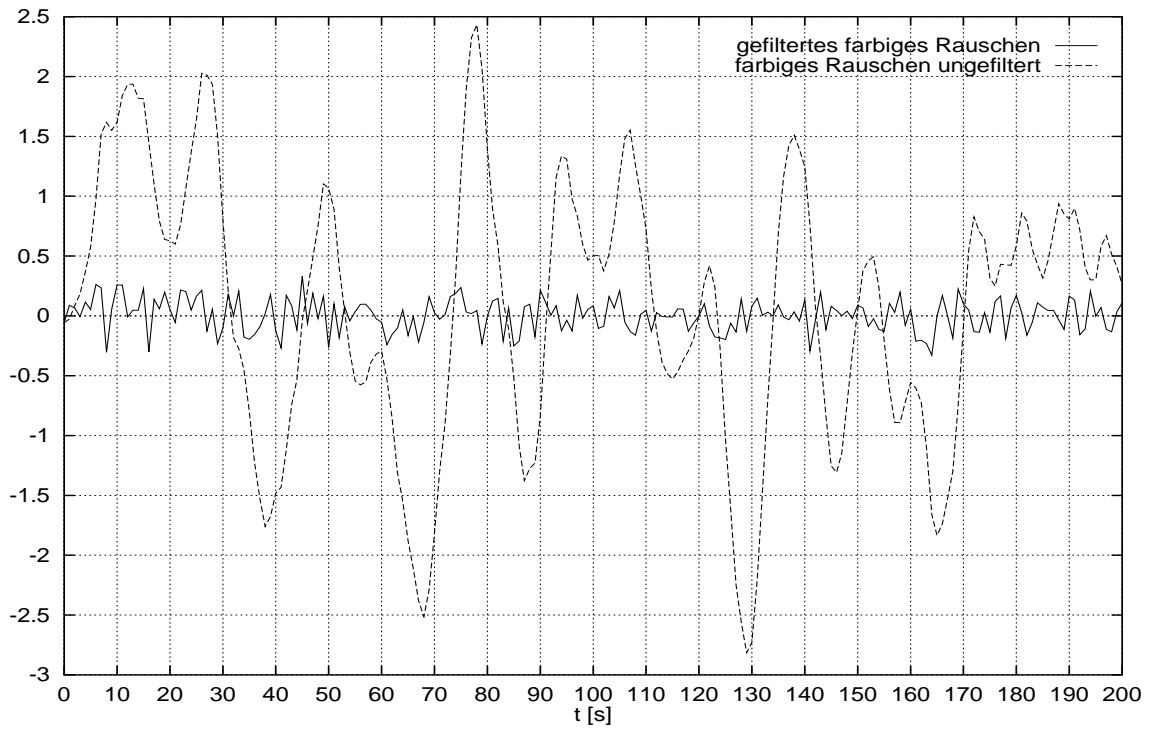
Vorhersagefehler-Filteroperatoren



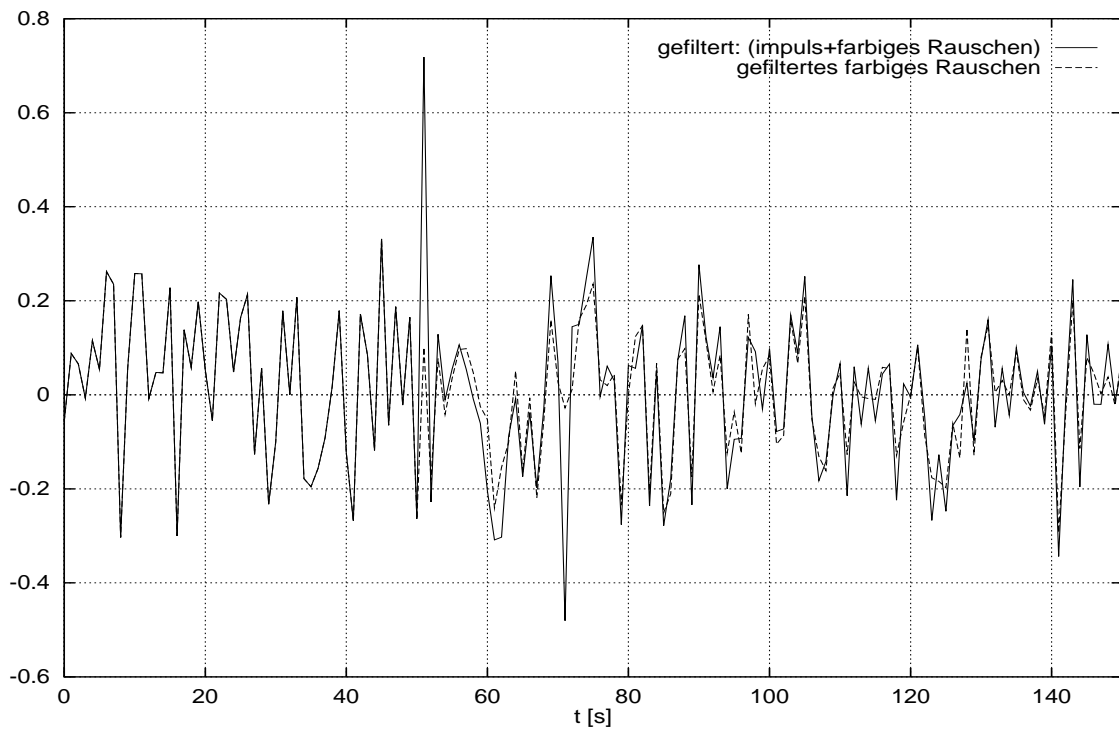
Spike-Dekonvolution durch Fehlervorhersage-Filterung

mit Vorhersagedistanz $\alpha = 1$

Fehlervorhersagefilterung des farbigen Rauschens



schwacher Sinusimpuls von farbigem Rauschen überlagert



Fehlervorhersagefilterung des farbig verrauschten Sinusimpulses

17 Zu den Programmen

17.1 Struktur des Verzeichnisbaumes

digsig/: Programme und Plots zu den Aufgaben 1 bis 16

digsig/aufgabe{#}/ : C-Programme zur Aufgabe {#}

digsig/aufgabe{#}/daten/ : Daten- und Plotfiles zur Aufgabe

17.2 Fast Fourier Transformation (FFT)

```
#include <math.h>
#define SWAP(a,b) tempr=(a);(a)=(b);(b)=tempr

void four1(float data[], unsigned long mn, int isign)
{
    unsigned long n,mmax,m,j,istep,i;
    double wtemp,wr,wpr,wpi,wi,theta;
    float tempr,tempi;
```

```

n=nn << 1;  j=1;
for (i=1;i<n;i+=2) {
  if (j > i) {
    SWAP(data[j],data[i]);  SWAP(data[j+1],data[i+1]);
  }
  m=n >> 1;
  while (m>= 2 && j > m) {
    j -= m;  m >>= 1;
  }
  j += m;
}
mmax=2;
while (n > mmax) {
  istep=mmax << 1;
  theta=isign*(6.28318530717959/mmax);
  wtemp=sin(0.5*theta);
  wpr = -2.0*wtemp*wtemp;
  wpi=sin(theta);
  wr=1.0;  wi=0.0;
  for (m=1;m<mmax;m+=2) {
    for (i=m;i<=n;i+=istep) {
      j=i+mmax;
      tempr=wr*data[j]-wi*data[j+1];  tempi=wr*data[j+1]+wi*data[j];
      data[j]=data[i]-tempr;          data[j+1]=data[i+1]-tempi;
      data[i] += tempr;              data[i+1] += tempi;
    }
    wr=(wtemp*wr)*wpr-wi*wpi+wr;      wi=wi*wpr+wtemp*wpi+wi;
  }
  mmax=istep;
}
}
#undef SWAP
/* (C) Copr. 1986-92 Numerical Recipes Software 50'0. */

```

17.3 Levinson Algorithmus

```

/* 1:1 Uebersetzung der Fortranroutine LEVIN (Skript, Seite 136)
*
* Variablen:
* float *r : Autokorrelierte des Eingangs (Laenge M)
*          baut die Toeplitzmatrix auf
* float *g : Kreuzkorrelierte von Eingang und gewünschtem Ausgang
*          (Laenge M)
* float *f : Wiener-Filter (Laenge M)

```

