

# Signalanalyse

## Übungsblatt 1

Nikolaus Hammler

16. Januar 2009

### Inhaltsverzeichnis

<b>1</b>	<b>2D LTI Systeme</b>	<b>2</b>
1.1	Aufgabenstellung und Signalflussgraphen . . . . .	2
1.2	Mögliche Rekursionsrichtungen . . . . .	3
1.3	Support der Impulsantwort . . . . .	4
1.4	Wahl der Startwerte . . . . .	4
<b>2</b>	<b>Stochastische Analyse</b>	<b>6</b>
2.1	Berechnung der ACF . . . . .	6
2.2	Berechnung der PSD . . . . .	7
2.3	Zufallsprozesse mit gleicher PSD . . . . .	8
<b>3</b>	<b>Geschwindigkeitsmessung</b>	<b>10</b>

# 1 2D LTI Systeme

## 1.1 Aufgabenstellung und Signalflussgraphen

In dieser Aufgabe geht es um eine einfache Analyse von LTI System mit zwei unabhängigen Variablen (2D). Es sollen folgende 2 LTI Systeme analysiert werden:

$$y[n_1, n_2] = x[n_1, n_2] + ay[n_1 - 1, n_2 - 1] + by[n_1, n_2 - 1] + cy[n_1 + 1, n_2 - 1] \quad (1)$$

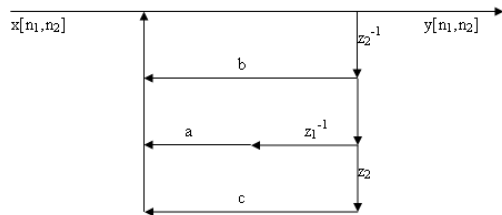
$$y[n_1, n_2] = x[n_1, n_2] + ay[n_1 - 1, n_2 - 1] + by[n_1, n_2 - 1] + cy[n_1 + 1, n_2] \quad (2)$$

Als erstes sollen nun die Signalflussgraphen gezeichnet werden. Um dies zu vereinfachen, werden Gleichungen 1 und 2 der  $z$ -Transformation unterworfen. Diese bilden den Signalfluss besser ab:

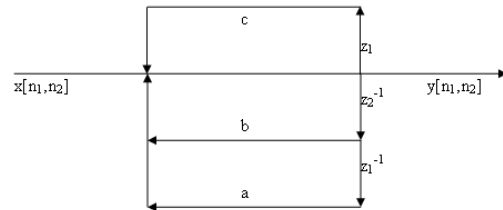
$$\begin{aligned} Y(z_1, z_2) &= X(z_1, z_2) + a \cdot Y(z_1, z_2) \cdot z_1^{-1} \cdot z_2^{-1} \\ &+ b \cdot Y(z_1, z_2) \cdot z_2^{-1} \\ &+ c \cdot Y(z_1, z_2) \cdot z_1 \cdot z_2^{-1} \end{aligned}$$

$$\begin{aligned} Y(z_1, z_2) &= X(z_1, z_2) + a \cdot Y(z_1, z_2) \cdot z_1^{-1} \cdot z_2^{-1} \\ &+ b \cdot Y(z_1, z_2) \cdot z_2^{-1} \\ &+ c \cdot Y(z_1, z_2) \cdot z_1 \end{aligned}$$

Daraus kann sofort der Signalflussgraph gezeichnet werden. Durch sortieren der  $z$  kann man eine kanonische Form finden (geringste Anzahl an Elementen). Für das erste System ist ein möglicher Graph in Abbildung 1(a) abgebildet, für das zweite System in Abbildung 1(b).



(a) System 1



(b) System 2

Abbildung 1: Signalflussgraphen

## 1.2 Mögliche Rekursionsrichtungen

Um die möglichen Rekursionsrichtungen zu bestimmen, wird der Signalflussgraph beispielhaft für ein Pixel in einem Bild aufgezeichnet, wobei die Breite mit  $n_1$  und die Höhe mit  $n_2$  gegeben ist. Der Graph ist direkt aus den Gleichungen 1 und 2 ersichtlich. Dabei wird jeweils das Pixel in der Mitte berechnet.

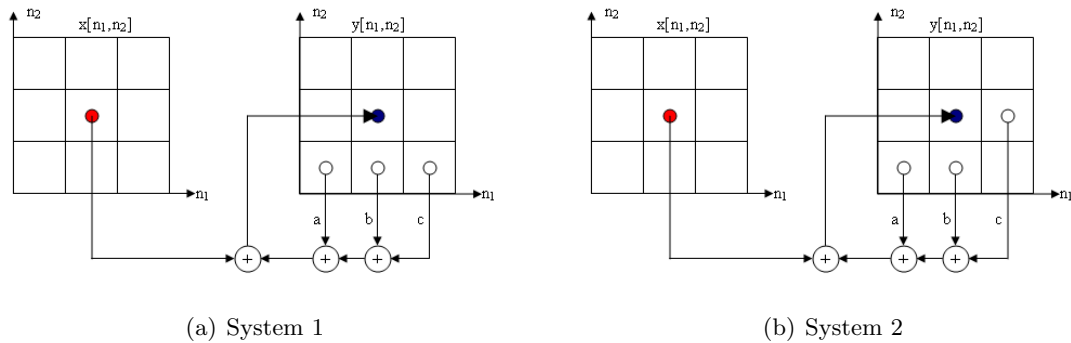


Abbildung 2: Signalflussgraphen für 3x3 Pixel

Die betreffenden Graphen sind in Abbildung 2(a) und 2(b) zu sehen. Eine Rekursion ist in solche Richtungen möglich, aus der keine Werte für die Rekursion bezogen werden. Das sind in Abbildung 2(a) (und damit System 1):

1. links
2. rechts
3. oben
4. oben-links
5. oben-rechts

Und für Abbildung 2(b) (System 2):

1. links
2. oben
3. links-oben
4. rechts-oben
5. rechts-unten

### 1.3 Support der Impulsantwort

Der Support der Impulsantwort ist jener Bereich, der von Null verschiedene Werte enthält. Daher muss die Impulsantwort bestimmt werden. Um diese zu bestimmen, wird ganz einfach der Einheitsimpuls  $\delta[n_1, n_2]$  als Eingangssignal  $x[n_1, n_2]$  gewählt und alle folgenden Pixel rekursiv bestimmt.

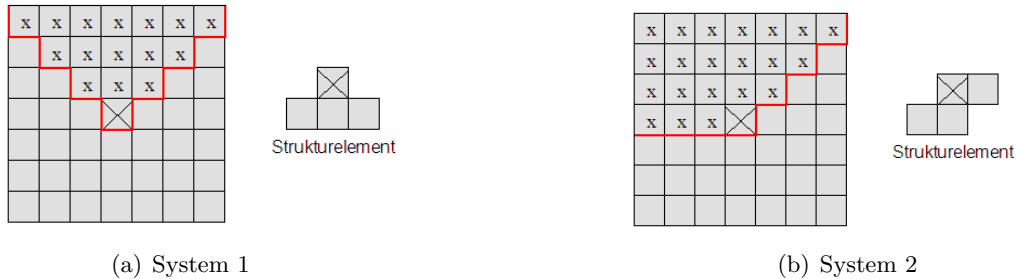


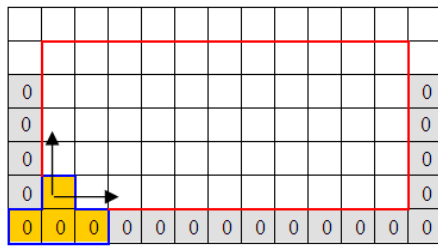
Abbildung 3: Support der Impulsantworten

Dies kann wie in Abbildung 3(a) und 3(b) durch eine Art Strukturelement veranschaulicht werden. Dabei repräsentiert das Raster das Bild wobei der Pixel in der Mitte den Einheitsimpuls darstellt. Das Eingangssignal ist also überall null ausser im Pixel in der Mitte. Das „Strukturelement“ fährt nun schrittweise über das Bild und das aktuelle Pixel unter dem Strukturelement (mit dem großen Kreuz) schaut, ob das Bild von irgendeinem anderen Pixel unter dem Strukturelement einen Support hat. Daraus ergeben sich die rot eingerahmten Bereiche im Bild als Support für die Impulsantwort.

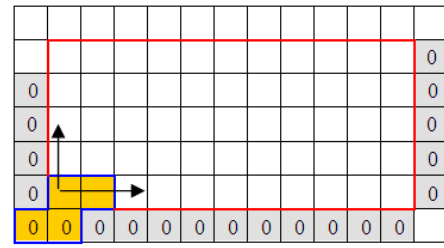
### 1.4 Wahl der Startwerte

Schließlich sollen die Startwerte für  $y[n_1, n_2]$  so gewählt werden, dass das System linear, verschiebungsvariant und berechenbar ist. Damit ein System *berechenbar* ist, müssen zu jeden Zeitpunkt alle benötigten Daten vorhanden sein. Das gilt insbesondere für die Startwerte. Um dies zu veranschaulichen ist in Abbildung 4(a) und 4(b) ein fiktives Bild (dicker, roter Rahmen gegeben). Wenn man nun mit den Strukturelementen aus dem vorigen Punkt über alle Pixel des Bildes geht, erkennt man, dass alle grauen Pixel ausserhalb des Bildes vorhanden sein müssen!

Zusätzlich stellt sich die Frage nach der Linearität und der Invarianz. Für die Linearität muss sichergestellt sein, dass für jedes beliebige Eingangssignal das Homogenitäts- und das Additivgesetz gilt. Für die Invarianz bedeutet dies, dass eine konstante Verschiebung am Eingang die gleiche Verschiebung am Ausgang hervorrufen muss. Insbesondere gilt das für die Startwerte. Sind nämlich die Werte in den grauen Pixel ungleich Null, so wird ein Bild erzeugt, obwohl kein Eingangssignal  $x[n_1, n_2]$  anliegt. Das verletzt natürlich das Linearitäts- und das Homogenitätsprinzip (ein lineares System muss als Ausgang die Nullfolge liefern, wenn die Eingangsfolge die Nullfolge ist).



(a) System 1



(b) System 2

Abbildung 4: Startwerte für die zwei Systeme

Es muss also sichergestellt sein, dass alle Startwerte im Anfangsruhezustand sind. Das ist genau dann gewährleistet, wenn alle grauen Pixel aus Abbildung 4(a) und 4(b) den Wert 0 haben.

## 2 Stochastische Analyse

### 2.1 Berechnung der ACF

Ein stochastischer Prozess sei durch Gleichung 3 beschrieben. Dabei bezeichnet  $Y$  eine Zufallsvariable. Die Phase  $Y$  ist zwischen  $-\pi$  und  $\pi$  gleichverteilt, d.h.  $Y \sim \mathcal{U}(-\pi, \pi)$ .

$$X[n] = a \cdot \sin(\omega_0 n + Y) \quad (3)$$

Als erstes soll die Autokorrelationsfolge  $\phi_{xx}[m, k]$  dieses Zufallsprozesses bestimmt werden. Die Autokorrelation (ACF) für diskrete Zufallsprozesse ist definiert als:

$$\phi_{XX}[m, k] = E\{X[m] \cdot X^*[k]\} \quad (4)$$

Für die Berechnung der ACF erweisen sich ausserdem folgende Definitionen als hilfreich:

$$\sin(x_1) \cdot \sin(x_2) = \frac{1}{2} (\cos(x_1 - x_2) - \cos(x_1 + x_2)) \quad (5)$$

$$E\{X\} = \int_{-\infty}^{\infty} x f_X(x) dx \quad (6)$$

Für die Berechnung fängt man einfach mit der Definition aus Gleichung 4 an:

$$\phi_{XX}[m, k] = E\{a \sin(\omega_0 m + Y) \cdot a \sin(\omega_0 k + Y)\}$$

Da es sich um ein reelles Signal handelt, kann die komplexe Konjugation (Gleichung 4) ignoriert werden. Mit Gleichung 5 folgt weiter:

$$\begin{aligned} \phi_{XX}[m, k] &= a^2 E \left\{ \frac{1}{2} [\cos(\omega_0 m + Y - \omega_0 k - Y) - \cos(\omega_0 m + Y + \omega_0 k + Y)] \right\} \\ &= \frac{a^2}{2} E \{ \cos(\omega_0 m - \omega_0 k) \} - \frac{a^2}{2} \{ \cos(\omega_0 m + \omega_0 k + 2Y) \} \\ &= \frac{a^2}{2} \cos(\omega_0(m - k)) - \frac{a^2}{2} \{ \cos(\omega_0(m + k) + 2Y) \} \end{aligned}$$

Es ist bemerkenswert, dass im ersten Term keine Zufallsvariable mehr vorkommt (d.h. der Term ist rein deterministisch), weshalb der Erwartungswert-Operator entfernt werden kann. Für den zweiten Term folgt aus Gleichung 6:

$$\begin{aligned}
E\{\cos(\omega_0(m+k) + 2Y)\} &= \int_{-\infty}^{+\infty} \cos(\omega_0(m+k) + 2y) f_Y(y) dy \\
&= \int_{-\pi}^{\pi} \cos(\omega_0(m+k) + 2y) \frac{1}{2\pi} dy \\
&= \frac{1}{2\pi} \left[ \frac{\sin(\omega_0(m+k) + 2y)}{2} \right]_{-\pi}^{\pi} \\
&= \frac{1}{4\pi} \underbrace{[\sin(\omega_0(m+k) + 2\pi) - \sin(\omega_0(m+k) - 2\pi)]}_0
\end{aligned}$$

Daraus folgt, dass der zweite Term sofort wegfällt und als Ergebnis übrig bleibt:

$$\phi_{XX}[m, k] = \frac{a^2}{2} \cos(\omega_0(m - k)) \quad (7)$$

Man erkennt sofort, dass der Wert dieses (ersten) Moments nur von der Differenz abhängig ist, das bedeutet, dass die ACF auch wie folgt geschrieben werden kann:

$$\phi_{XX}[\tau] = \frac{a^2}{2} \cos(\omega_0\tau) \quad (8)$$

Damit dieser Zufallsprozess allerdings (zumindest) schwachstationär ist, müsste noch bewiesen werden, dass die Momente zweiter Ordnung (Varianzen) ebenfalls nur von der Differenz abhängig sind.

## 2.2 Berechnung der PSD

Für das nächste Beispiel kann diese Stationarität jedoch angenommen werden. Nun soll von diesem Zufallsprozess nämlich analytisch die PSD (Power Spectrum Density) bestimmt werden. Dafür ist das Wiener Khinchin Theorem (Gleichung 9) sehr nützlich:

$$P_{XX}(\omega) = \sum_{k=-\infty}^{\infty} \phi_{XX}[k] e^{-j\omega k} \quad (9)$$

Das Wiener Khinchin Theorem stellt den Zusammenhang zwischen der ACF und der PSD eines (mittelwertfreien, stationären!) Zufallsprozesses mit Hilfe der DTFT her. Aus den Vorlesungsunterlagen lässt sich die Fourier Transformierte der DTFT für ein cos-Signal herauslesen:

$$\mathcal{F}\{\cos(\omega_0 n + \phi)\} = \pi \sum_{k=-\infty}^{\infty} [e^{j\phi} \delta(\omega - \omega_0 + 2\pi k) + e^{-j\phi} \delta(\omega + \omega_0 + 2\pi k)] \quad (10)$$

Gleichung 8 in Gleichung 10 eingesetzt ergibt ( $\phi = 0$ ):

$$P_{XX}(\omega) = \frac{a^2\pi}{2} \sum_{k=-\infty}^{\infty} [\delta(\omega - \omega_0 + 2\pi k) + \delta(\omega + \omega_0 + 2\pi k)] \quad (11)$$

Betrachtet man nur einen Ausschnitt des (sich unendlich wiederholenden) Spektrums von 0 bis  $2\pi$ , so erhält man die gewohnten Peaks bei der Frequenz  $\omega_0$  und seiner Spiegelfrequenz  $2\pi - \omega_0$ :

$$P_{XX}(\omega) = \frac{a^2\pi}{2} [\delta(\omega - \omega_0) + \delta(\omega + \omega_0 + 2\pi)] \quad (12)$$

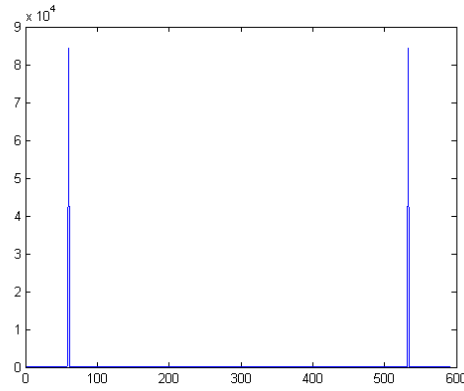


Abbildung 5: PSD Plot für den Zufallsprozess

Abbildung 5 zeigt einen MATLAB-Plot für die PSD in Gleichung 12 (Achtung, Leakage-Effekt durch Fensterung!).

### 2.3 Zufallsprozesse mit gleicher PSD

Im letzten Beispiel sollen zwei weitere (möglichst unterschiedliche) Zufallsprozesse gefunden werden, die die gleiche PSD erzeugen wie Gleichung 11. Dazu ist eine weitere Definition der PSD aus dem deterministischen Spektrum hilfreich (das quadrierte Betragsspektrum):

$$P_{XX}(\omega) = X(\omega) \cdot X^*(\omega) = |X(\omega)|^2 \quad (13)$$

Eine erste Klasse von Zufallsprozessen kann über folgende zwei Beobachtungen gefunden werden:



1. Ein Sinussignal mit beliebiger Phase erzeugt immer eine Autokorrelationsfunktion der Form  $\cos(\omega\tau)$ . Die Phaseninformation geht verloren. Da mit Gleichung 9 eine bijektive Abbildung zur PSD gegeben ist, genügt es Zufallsprozesse zu finden, die die ACF  $\cos(\omega_0\tau)$  generieren.
2. Anhand Gleichung 13 lässt sich erkennen, dass für die PSD nur der Betrag relevant ist, auch hier geht die Phaseninformation komplett verloren.

Beide Punkte zeigen, dass die Bestimmung - sowohl der ACF als auch der PSD - nicht eindeutig ist und mit einem **Datenverlust** verbunden ist. Es erzeugen also alle Zufallsprozesse mit zufällig verteilter Phase als auch alle deterministischen Funktionen die genau eine Frequenz  $\omega_0$  mit Amplitude  $a$  enthalten, die gleiche PSD wie in Gleichung 11. Der Kreativität für solche Zufallsprozesse sind also kaum Grenzen gesetzt:

- $X[n] = a \cdot \sin(\omega_0 n + Y) \quad Y \sim \mathcal{N}(\mu, \sigma^2)$
- $X[n] = a \cdot \sin(\omega_0 n + Y_1 + Y_2) \quad Y_1 \sim \mathcal{N}(\mu, \sigma^2) \quad Y_2 \sim \mathcal{U}(-2, \pi)$
- $X[n] = -a \cdot \sin(\omega_0 n + Y)$
- $X[n] = a \cdot \cos(\omega_0 n + Y)$
- ...

Eine andere interessante Möglichkeit ergibt sich wenn man sich überlegt, dass auch komplexe Signals das gleiche Betragsspektrum liefern können. Beispielsweise ist das der Zufallsprozess

$$X[n] = a \cdot \sin(\omega_0 n + Y) + j \cdot a \sin(\omega_0 n + Y) \quad Y \sim \mathcal{U}(-\pi, \pi)$$

### 3 Geschwindigkeitsmessung

Im letzten Beispiel soll die Geschwindigkeit einer Flüssigkeit mittels LDA Verfahren bestimmt werden. Wir haben dazu einen Datenvektor des LDA Systems erhalten. Von diesen Daten soll nun die PSD mit MATLAB geschätzt werden und dazu eine Schätzfunktion geschrieben werden, die eine Periodogrammmittelung durchführt.

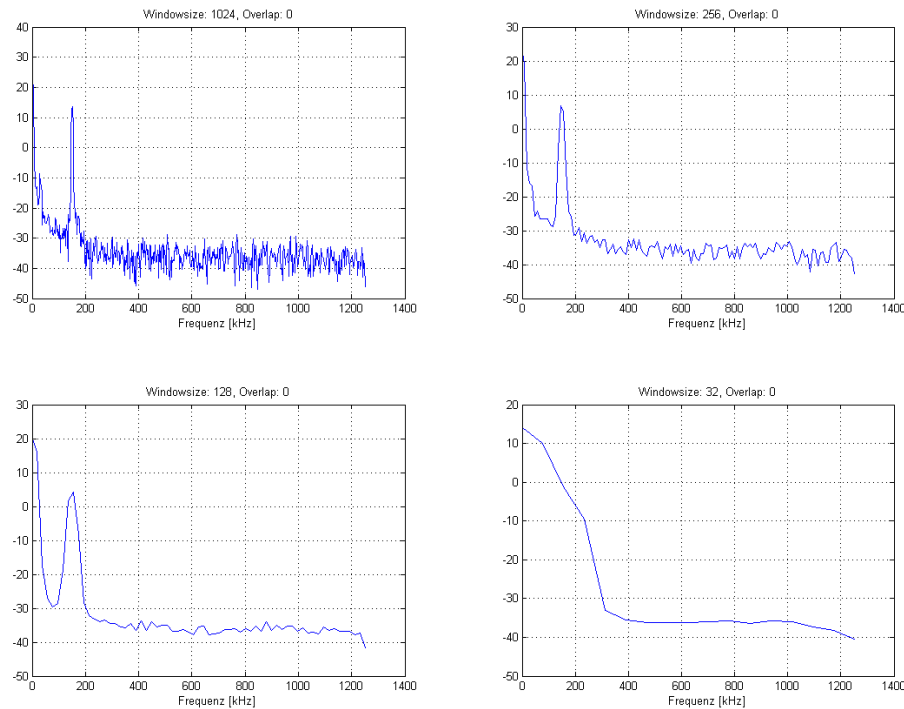


Abbildung 6: Gewichtung mit Hammingfenster

Es gibt eine große Auswahl an Fenstern. Die wesentlichen Unterschiede zwischen ihnen sind Breite des Hauptmaximums (soll so dünn wie möglich sein) und Höhe der Nebenkeule (soll so niedrig wie möglich sein). Die Breite des Hauptmaximums ist für die Frequenzauflösung verantwortlich, die Höhe der Nebenkeule für den Leakagefaktor. Zum Test habe ich zwei konträre Fenster gewählt:

- Das Rechteckfenster eine sehr hohe Frequenzauflösung, dafür einen maximalen Leakagefaktor
- Das Hammingfenster hat eine sehr niedrige Frequenzauflösung, durch die niedrigen Nebenkeulen jedoch einen geringen Leakagefaktor

Für diese Anwendung ist die Amplitude eher nebensächlich, viel wichtiger ist es, eine maximale Frequenz eindeutig herauslesen zu können. Der Leakagefaktor darf nicht so

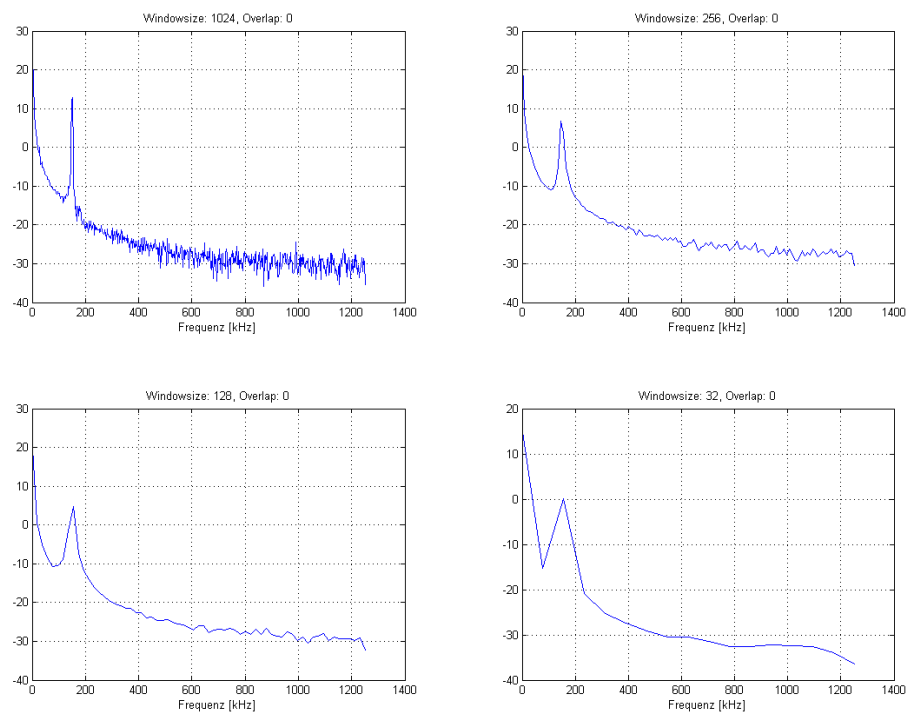


Abbildung 7: Gewichtung mit Rechteckfenster

hoch werden, dass Frequenzen entstehen, die die echte Frequenz überdecken. Umgekehrt darf die Frequenzauflösung nicht so niedrig werden, dass dadurch die Frequenz falsch bestimmt wird.

Abbildung 7 zeigt den PSD Schätzer für variable Blockgrößen, wobei das Rechteckfenster verwendet wurde. Die PSD stelle ich prinzipiell logarithmisch dar, um einen besseren Überblick zu bekommen. Als Blockgröße habe ich generell nur Zweierpotenzen gewählt, da durchs Zeropadding ohnehin nur zusätzliche Fehler entstehen würden. Deutlich zu erkennen man, dass mit Anzahl der Mittelungen die Varianz der PSD abnimmt. Beim Rechteckfenster bleiben die Peaks deutlich sichtbar, sogar bei einer Blockgröße von nur 32 Samples.

Abbildung 6 zeigt das fast gegenteilige Fenster - das Hammingfenster. Im Gegensatz zu Abbildung 7 fallen hier sofort zwei Dinge auf, die vor allem mit größerer Anzahl an Mittelungen deutlicher werden: Das Rechteckfenster erzeugt deutlich mehr Leakage. Das ist am besten bei der Blockgröße 128 erkennbar: Das Rechteckfenster hat zwischen den beiden Peaks deutlich mehr Leakage. Andererseits erkennt man das Problem mit der Frequenzauflösung sehr gut: Während man die zwei Peaks beim Rechteckfenster noch gut erkennen kann, verschwimmen sie beim Hammingfenster derart, dass sie nicht mehr auseinander zu kennen sind.

Die Frage nach der geeigneten Fensterfunktion ist also schwer zu beantworten. Da der gesuchte Wert ja nur proportional zu unserer gesuchten Frequenz ist, ist die Höhe nicht von Bedeutung, andererseits ist das Signal sehr schmalbandig. Für den Peak haben jedenfalls alle Verfahren in etwa den gleichen Wert geliefert. Ich habe mich jedoch für das Rechteckfenster entschieden: Die Leakagefrequenzen stören nicht zu sehr, dafür gibt es eine gute Frequenzauflösung.

In weiterer Folge sollen unterschiedliche Überlappungen zwischen 0% und 50% getestet werden.

Die Ergebnisse sind in Abbildung 8 und 9 zu sehen. Es wurde eine Überlappung von 50% gewählt. Der Unterschied zu Abbildung 6 und 7 ist sehr marginal. Der Graph wurde lediglich etwas mehr geglättet, was wahrscheinlich auf die erhöhte Anzahl von Mittelungen zurückzuführen ist. Selbstverständlich habe ich auch andere Überlappungen durchprobiert, da der Unterschied allerdings nicht merklich zu erkennen war, habe ich diese Plots nicht ausgedruckt.

Als nächstes soll das niederfrequente Rauschsignal entfernt werden. Das Rauschsignal befindet sich sehr niederfrequent nahe des Gleichanteils während das Nutzsignal bei ca. 150 kHz liegt. Deshalb wird ein Hochpassfilter benötigt. Es wurde ein Cauer-Filter gewählt, da dieser eine maximale Steigung besitzt. Die Ripples im Durchlassbereich sind egal, dafür werden möglichst viel niederfrequente Störungen rausgefiltert. Die Grenzfrequenz wurde auf 110 kHz gesetzt.

Die bisherige Aufgabe soll sodann für das bereinigte Signal wiederholt werden. Die Ergebnisse sind in Abbildung 10 bis 13 zu sehen.

Bei allen Ergebnissen ist der verbleibende Peak für das Nutzsignal gut ausgebildet. Für die Bestimmung der eigentlichen Geschwindigkeit wurden jedoch folgende Parameter gewählt:

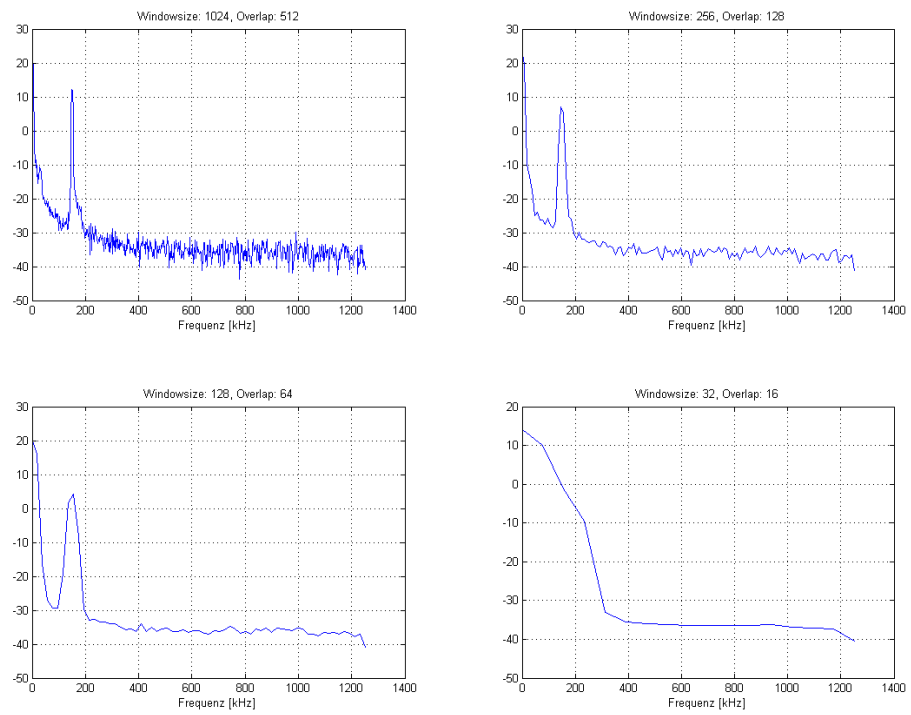


Abbildung 8: Gewichtung mit Hammingfenster und Überlappung

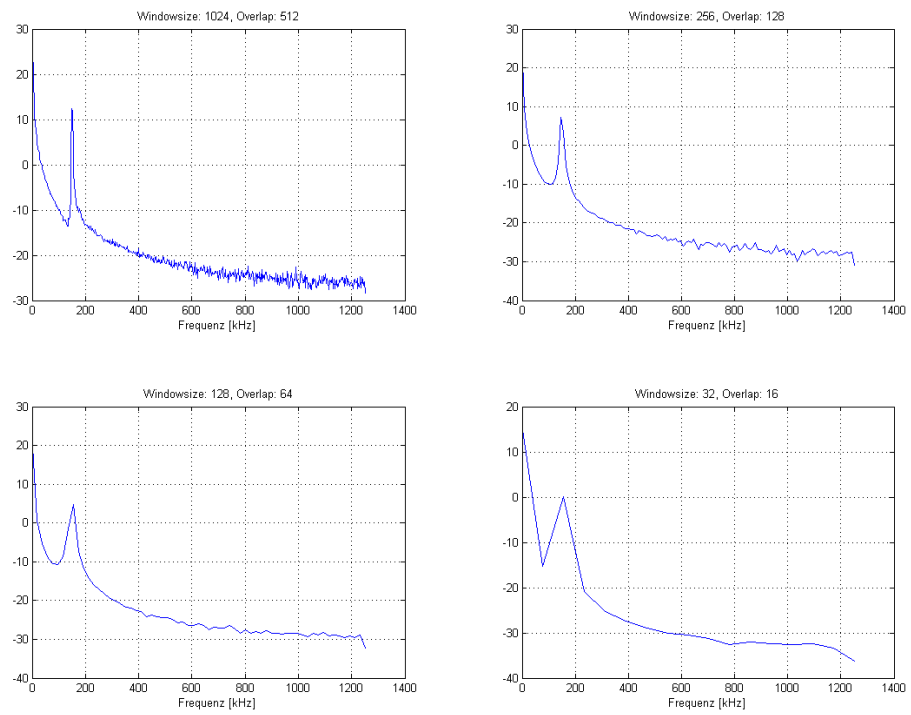


Abbildung 9: Gewichtung mit Rechteckfenster und Überlappung

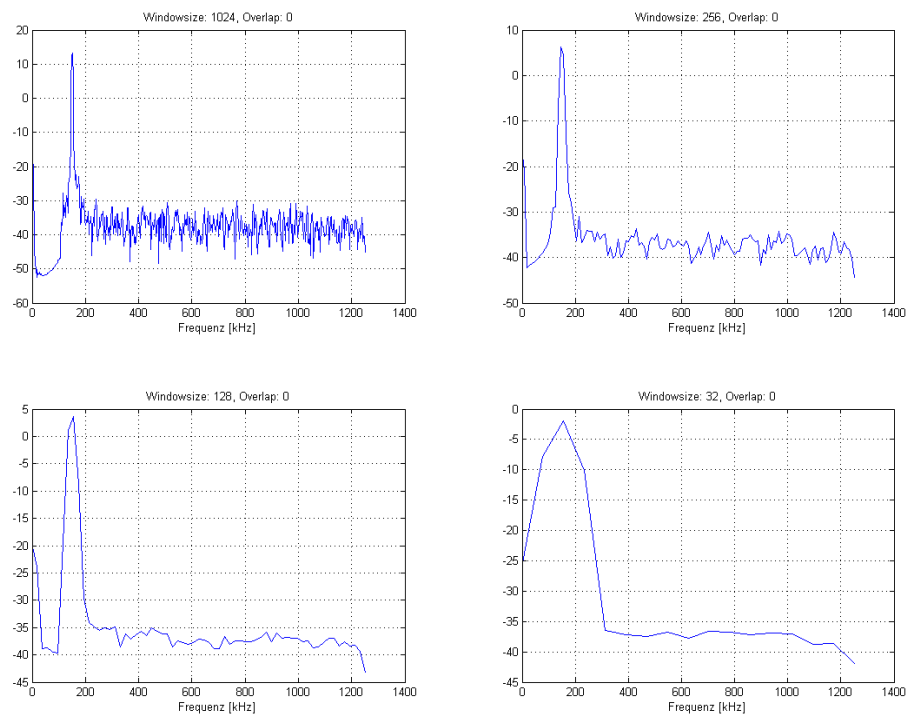


Abbildung 10: Ausgabe: Hammingfenster ohne Overlap

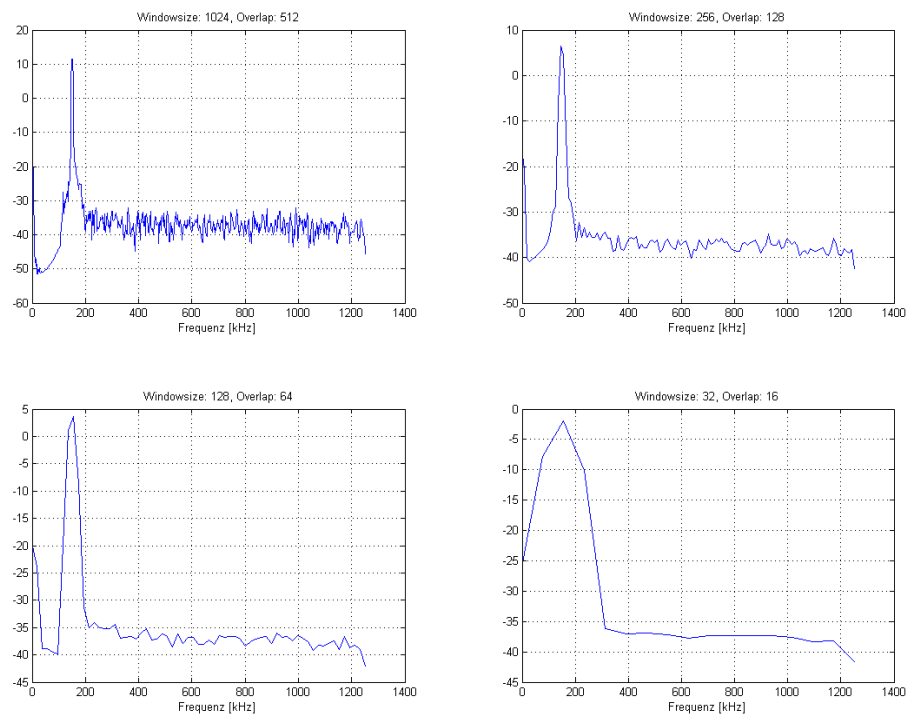


Abbildung 11: Ausgabe: Hammingfenster mit Overlap



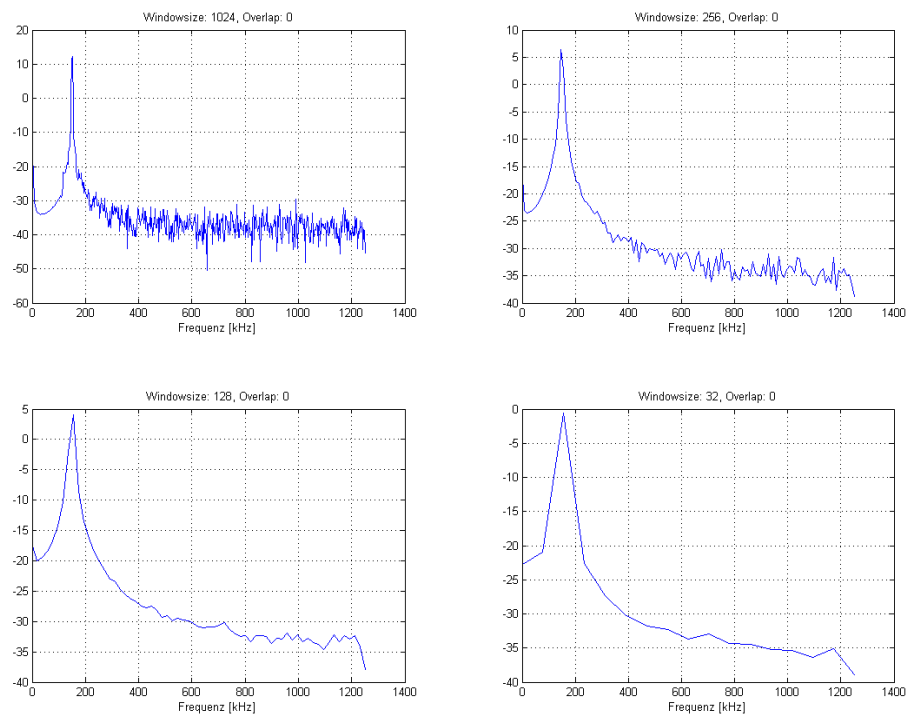


Abbildung 12: Ausgabe: Rechteckfenster ohne Overlap

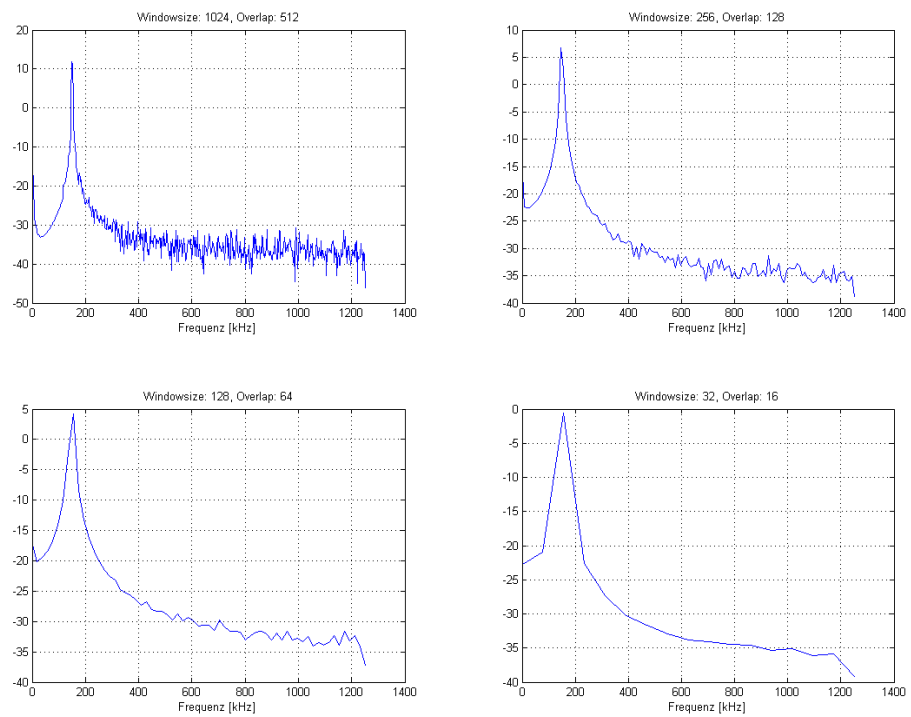


Abbildung 13: Ausgabe: Rechteckfenster mit Overlap

- Fenster: Rechteck. Der Peak ist hier am besten ausgeprägt (am spitzesten) und der langsame Abfall durch allfälliges Leakage stört nicht weiter.
- Overlap: 50%. Dadurch wird zusätzlich eine bessere Glättung erreicht
- Fensterlänge: 256 Samples. Dadurch kann genügend oft gemittelt werden, dennoch bleiben genug Frequenzen bei der Darstellung übrig. Viel kleiner sollte dieser Wert nicht mehr gesetzt werden, denn dadurch sinkt auch die Anzahl der darstellbaren Frequenzen und damit steigt die Ungenauigkeit.

Mit diesen Werten wurde eine Frequenz von 155,07 kHz ermittelt. Dieser Wert muss lediglich noch mit folgender Konstante multipliziert werden:

$$d = \frac{\lambda}{2 \sin(\theta/2)}$$

Der damit ermittelte Schätzwert entspricht einer Geschwindigkeit von **0,1893** Meter pro Sekunde.