4. Spektralanalyse

- Die Spektralanalyse ermittelt, welche Beiträge die einzelnen Frequenzen zu einem Signal liefern.
- Je nach Art des Zeitsignals wird der Frequenzgehalt durch die Fourier-Transformation, die Fourier-Reihe oder das Leistungsdichtespektrum beschrieben.
- Mithilfe der diskreten Fourier-Transformation lassen sich in allen drei Fällen Abschätzungen für den Frequenzgehalt aus endlichen Zeitreihen berechnen.

4. Spektralanalyse

Signal		Spektrum
transient →	Abtasten + FFT	→ Fourier-Transformation
periodisch →		→ Fourier-Reihe
stochastisch →		→ Leistungsdichtespektrum

4. Spektralanalyse

4.1 Transiente Signale

4.2 Periodische Signale

4.3 Stochastische Signale

4.1 Transiente Signale

- Der Frequenzgehalt von transienten Signalen wird durch ihre Fourier-Transformation beschrieben.
- Aufgabenstellung:
  - Gegeben ist die Zeitreihe $x_n = x(n\,\Delta t), \quad n = 0, \ldots, N-1$, die durch Abtasten mit der Abtastrate f_S aus dem transienten Signal $x(t)$ gewonnen wurde.
  - Es wird vorausgesetzt, dass die Fourier-Transformation des Zeitsignals existiert und beim Abtasten die Nyquist-Bedingung $f_A = f_S/2 \geq f_C$ eingehalten wurde.
  - Mithilfe der Zeitreihe soll eine Abschätzung für die Fourier-Transformation gefunden werden.

4.1 Transiente Signale

- Abschätzung der Fourier-Transformation:
  - Nach der Rechteckregel wird das Fourier-Integral approximiert durch

$$\hat{X}(f) = \sum_{n=0}^{N-1} x_n e^{-2\pi i f n \Delta t} \Delta t, \quad -f_A \leq f \leq f_A$$

  - Auswertung für die Frequenzen

$$f_k = \frac{k}{N} f_S = \frac{k}{N \Delta t}, \quad -\frac{N}{2} \leq k \leq \frac{N}{2}$$

ergibt: $$\hat{X}(f_k) = \Delta t \sum_{n=0}^{N-1} x_n e^{-2\pi i k n / N} = \Delta t \, X_k$$

4.1 Transiente Signale

- Wegen $X_{-N/2} = X_{N/2}$ ergeben sich nur N verschiedene Werte.
- Die Indizes $0 \le k \le N/2$ entsprechen den positiven Frequenzen $0 \le f_k \le f_A$.
- Für die Frequenzauflösung gilt:

$$\Delta f = f_{k+1} - f_k = \frac{f_S}{N} = \frac{1}{N\,\Delta t} = \frac{1}{T}$$

- Die Frequenzauflösung ist umgekehrt proportional zur Anzahl der Messwerte und umgekehrt proportional zur Messdauer T.
- Die Frequenzauflösung lässt sich erhöhen, indem die Zeitreihe durch Anfügen von Nullen verlängert wird (Zero padding).

4.1 Transiente Signale

- Inverse Transformation:
  - Da das Zeitsignal nach Voraussetzung nur Frequenzen mit $|f| \leq f_A$ enthält, gilt:

$$x(t) = \int_{-f_A}^{f_A} X(f)\, e^{2\pi i f t}\, df$$

  - Approximation nach der Rechteckregel ergibt:

$$\hat{x}(t) = \sum_{k=-N/2}^{N/2-1} \hat{X}(f_k)\, e^{2\pi i k \Delta f t}\, \Delta f = \sum_{k=-N/2}^{N/2-1} X_k\, \Delta t\, e^{2\pi i k t/T} \frac{1}{N\,\Delta t}$$

$$= \frac{1}{N} \sum_{k=-N/2}^{N/2-1} X_k\, e^{2\pi i k t/T}$$

4.1 Transiente Signale

- Für $t = n\Delta t$ gilt:

$$\hat{x}(n\Delta t) = \frac{1}{N}\sum_{k=-N/2}^{N/2-1} X_k e^{2\pi i k n \Delta t/T} = \frac{1}{N}\sum_{k=0}^{N-1} X_k e^{2\pi i k n/N} = x_n$$

- Für die letzte Umformung wurde $X_{-k} = X_{N-k}$ benutzt.
- Die Fourier-Reihe

$$\hat{x}(t) = \frac{1}{N}\sum_{k=-N/2}^{N/2-1} X_k e^{2\pi i k t/T}$$

  interpoliert im Zeitbereich $0 \leq t \leq T$ die Zeitreihe und setzt sich außerhalb dieses Intervalls periodisch fort.

4.1 Transiente Signale

- Ergebnis:
  - Zwischen der diskreten Fourier-Transformation der Zeitreihe und der Fourier-Transformation des Zeitsignals bestehen folgende Zusammenhänge:

$$\hat{X}\left(f_S \frac{k}{N}\right) = \Delta t\, X_k\,, \quad x_n = \frac{1}{N\,\Delta t} \sum_{k=0}^{N-1} \hat{X}\left(f_S \frac{k}{N}\right) e^{2\pi i k n/N}$$

  - Für die Berechnung mit GNU Octave können die Funktionen **`fft`** und **`ifft`** verwendet werden.

4.1 Transiente Signale

- Beispiel:
  - Für den Impuls

$$x(t) = \begin{cases} \sin^2\left(\pi \dfrac{t}{T}\right), & 0 \leq t \leq T \\ 0, & t > T \end{cases}$$

    soll die Fourier-Transformation mit GNU Octave berechnet werden.
  - Dabei soll der Einfluss der Abtastrate untersucht werden.
  - Zahlenwert: $T = 1\ \mathrm{ms}$

4.1 Transiente Signale

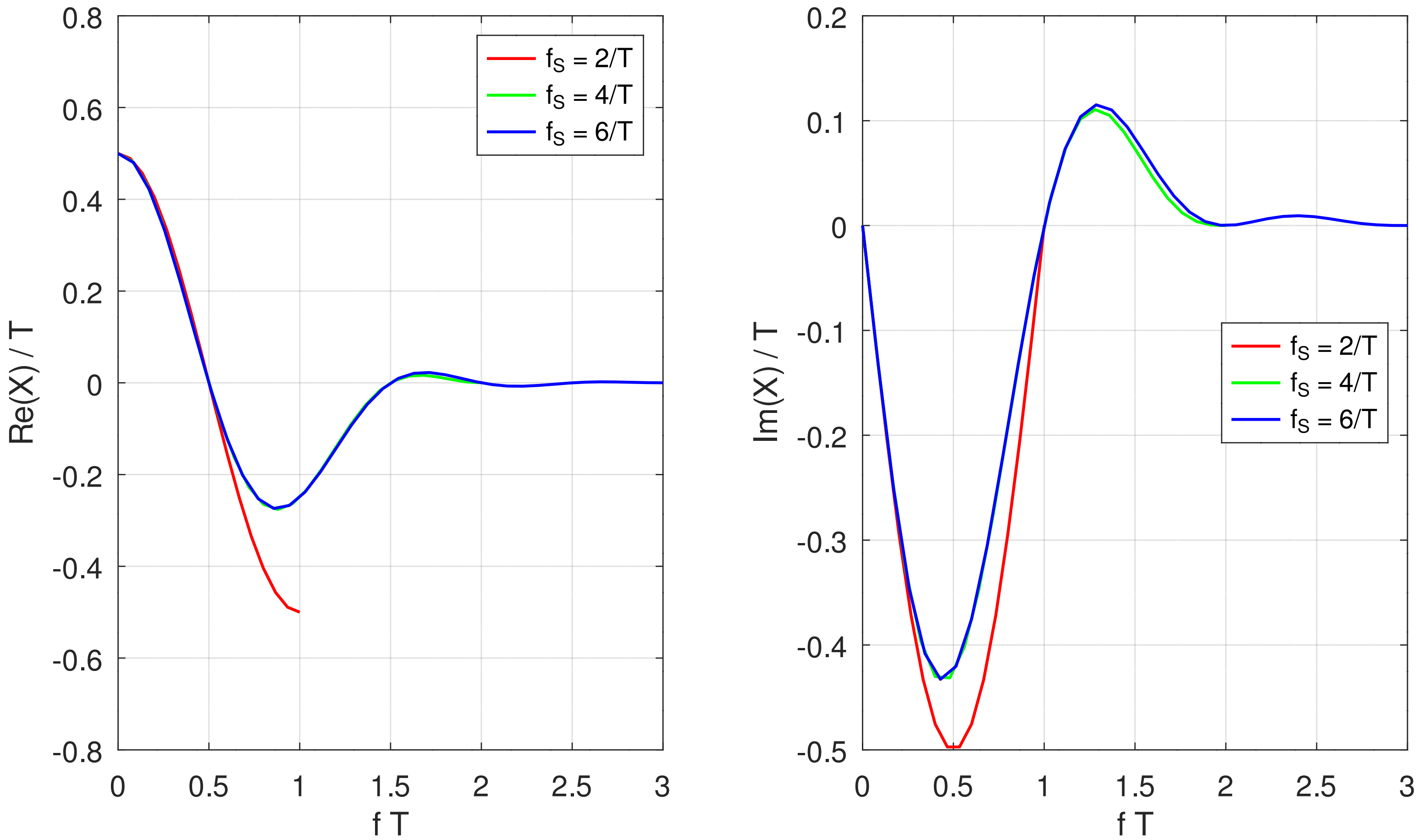
- GNU Octave-Skript:

```
# Daten

  T  =          0.001;  % Impulsdauer
  n  =             10;  % Messdauer = n * T
  fS = [2, 4, 6] / T;   % Abtastraten

# Rechnung

  X = cell(3, 1); f = cell(3, 1);

  for m = 1 : 3
      dt = 1 / fS(m);
      t  = 0 : dt : T;
      x  = sin(pi * t / T).^2;              % Impuls
      N  = n * length(t); N += mod(N, 2); % N gerade
      Xd = fft(x, N);
      X{m} = dt * Xd(1 : N/2 + 1) / T;      % X/T, k = 0, ..., N/2
      f{m} = T * (0 : N/2) * fS(m) / N;     % T * f
      labels{m} = sprintf("f_S = %d/T", fs(m) * T);
  end
```

4.1 Transiente Signale

```
# Ausgabe

  figure(1, "position", [100, 100, 800, 400],
            "paperposition", [0, 0, 24, 13]);
  subplot(1, 2, 1);
     plot(f{1}, real(X{1}), "color", "red",
          f{2}, real(X{2}), "color", "green",
          f{3}, real(X{3}), "color", "blue");
     legend(labels);
     xlim([0, 0.5 * max(fS) * T]); grid;
     xlabel("f T"); ylabel("Re(X) / T");
  subplot(1, 2, 2);
     plot(f{1}, imag(X{1}), "color", "red",
          f{2}, imag(X{2}), "color", "green",
          f{3}, imag(X{3}), "color", "blue");
     legend(labels, "location", "east");
     xlim([0, 0.5 * max(fS) * T]); grid;
     xlabel("f T"); ylabel("Im(X) / T");

  print([file, ".svg"], "-dsvg");
```

4.1 Transiente Signale



4.2 Periodische Signale

- Der Frequenzgehalt von periodischen Signalen wird durch die Koeffizienten der Fourier-Reihe beschrieben.
- Komplexe Fourier-Reihe:
  - Die reelle Fourier-Reihe eines Zeitsignals der Periode T ist:

$$x(t)=c_0+\sum_{k=1}^{\infty}\left(c_k\cos\left(2\pi k\frac{t}{T}\right)+s_k\sin\left(2\pi k\frac{t}{T}\right)\right)$$

  - Für die trigonometrischen Funktionen gilt:

$$\cos(2\pi k t/T)=\frac{1}{2}\left(e^{2\pi i k t/T}+e^{-2\pi i k t/T}\right)$$

$$\sin(2\pi k t/T)=\frac{1}{2i}\left(e^{2\pi i k t/T}-e^{-2\pi i k t/T}\right)$$

4.2 Periodische Signale

- Einsetzen ergibt:

$$x(t)=c_0+\frac{1}{2}\sum_{k=1}^{\infty}\left[(c_k-i\,s_k)\,e^{2\pi i k t/T}+(c_k+i\,s_k)\,e^{-2\pi i k t/T}\right]$$

- Mit $C_k=c_k-i\,s_k\,,\;\; C_0=2\,c_0\,,\;\; C_{-k}=\bar{C}_k$

  vereinfacht sich die Reihe zu: $x(t)=\frac{1}{2}\sum_{k=-\infty}^{\infty}C_k\,e^{2\pi i k t/T}$

- Für die zugehörigen Frequenzen gilt: $f_k=k/T$

4.2 Periodische Signale

- Periodische Zeitreihe:
  - Abtasten einer Periode des Zeitsignals mit dem Zeitschritt $\Delta t = T/N$ ergibt:

$$x_n = x(n\Delta t) = \frac{1}{2}\sum_{k=-\infty}^{\infty} C_k e^{2\pi i k n \Delta t/T}, \quad n = 0, \ldots, N-1$$

  - Mit $\Delta t / T = 1/N$ folgt:

$$x_n = \frac{1}{2}\sum_{k=-\infty}^{\infty} C_k e^{2\pi i k n/N}, \quad n = 0, \ldots, N-1$$

  - Wegen

$$e^{2\pi i (k+N) n/N} = e^{2\pi i k n/N} e^{2\pi i n} = e^{2\pi i k n/N}$$

    enthält die unendliche Reihe nur N verschiedene Werte der komplexen Exponentialfunktion.

4.2 Periodische Signale

- Sie lässt sich daher als endliche Summe schreiben:

$$x_n = \frac{1}{2} \sum_{k=-N/2}^{N/2-1} \tilde{C}_k e^{2\pi i k n/N}, \quad n = 0, \ldots, N-1$$

$$\text{mit} \quad \tilde{C}_k = \sum_{l=-\infty}^{\infty} C_{k+lN}$$

- Die größte enthaltene Frequenz ist:

$$f_{max} = f_{N/2} = \frac{N}{2} \frac{1}{T} = \frac{N}{2} \frac{1}{N \Delta t} = \frac{1}{2 \Delta t} = \frac{f_S}{2} = f_A$$

- Wenn die Nyquist-Bedingung erfüllt ist, gilt: $\tilde{C}_k = C_k$

4.2 Periodische Zeitsignale

- Zusammenhang mit der diskreten Fourier-Transformation:
  - Mit den Werten X_k der diskreten Fourier-Transformation der Zeitreihe gilt:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{2\pi i k n/N} = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} X_k e^{2\pi i k n/N}$$

  - Daraus folgt:

$$C_k = c_k - i s_k = \frac{2}{N} X_k \rightarrow c_k = \frac{2}{N} \Re\left(X_k\right), \quad s_k = -\frac{2}{N} \Im\left(X_k\right)$$

  - Die angegebenen Beziehungen gelten auch dann, wenn die Länge des Zeitsignals ein ganzzahliges Vielfaches der Periode ist.

4.2 Periodische Signale

- Beispiel:
  - Betrachtet wird eine periodische Funktion mit der Periode $T = 0{,}1\ \mathrm{s}$.
  - Die Koeffizienten der Fourier-Reihe haben die Werte $s_1 = 2$, $c_3 = 5$, $s_4 = -1{,}5$, $c_4 = 2$. Alle anderen Koeffizienten sind null.
  - Die zugehörigen Frequenzen sind $f_1 = 10\ \mathrm{Hz}$, $f_3 = 30\ \mathrm{Hz}$ und $f_4 = 40\ \mathrm{Hz}$.
  - Das folgende Skript erzeugt eine Zeitreihe mit zwei Perioden und berechnet daraus die Koeffizienten der Fourier-Reihe.

4.2 Periodische Signale

```
# Daten

  T  = 0.1;                           % Periode
  f1 = 10; s1 = 2;                    % Frequenz 1
  f3 = 30; c3 = 5;                    % Frequenz 2
  f4 = 40; s4 = -1.5; c4 = 2          % Frequenz 3
  nT =  2;                            % Anzahl Perioden

# Zeitreihe

  fS = 2.5 * f4;                      % Abtastrate
  dt = 1 / fS;                        % Zeitschritt
  N  = nT * T * fS;                   % Anzahl Messpunkte
  t  = (0 : N - 1) * dt;              % Zeitpunkte
  pt = 2 * pi * t;
  x  = s1 * sin(f1 * pt) + c3 * cos(f3 * pt);
  x += s4 * sin(f4 * pt) + c4 * cos(f4 * pt);

# Fourier-Transformation

  C = 2 * fft(x) / N;  C(1)= 0.5 * C(1);
  f = (0 : N/2) / (nT * T);
```

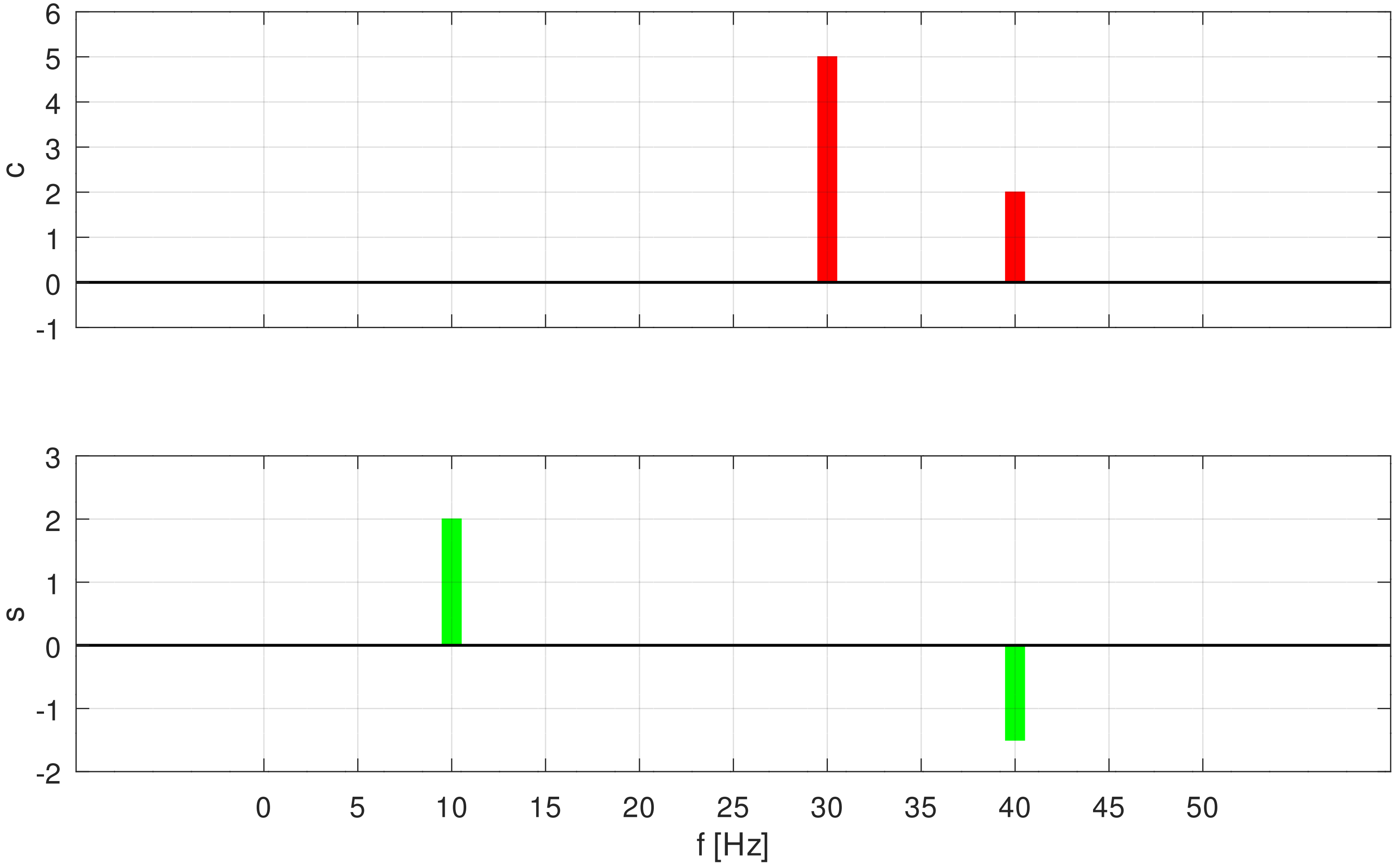
4.2 Periodische Signale

```
  c =  real(C(1 : N/2 + 1));
  s = -imag(C(1 : N/2 + 1));

# Ausgabe

  figure(1, "position", [100, 500, 1000, 800],
            "paperposition", [0, 0, 25, 14]);
  subplot(2, 1, 1);
     bar(f, c, "barwidth", 0.2, "edgecolor", "r",
                                "facecolor", "r");
     grid;
     axis("labely");
     ylabel("c");
  subplot(2, 1, 2, "position", [0.13, 0.17, 0.775,0.34]);
     bar(f, s, "barwidth", 0.2, "edgecolor", "g",
                                "facecolor", "g");
     grid;
     xlabel("f [Hz]");
     ylabel("s");

  print("v3_4_2.svg", "-dsvg");
```

4.2 Periodische Signale

- Das Bild auf der folgenden Seite zeigt, dass die Koeffizienten der Fourier-Reihe korrekt identifiziert und ihren Frequenzen zugeordnet werden.
- Die Methode kann verwendet werden, um die Fourier-Koeffizienten von analytisch gegebenen Funktionen zu ermitteln.

4.2 Periodische Signale



4.2 Periodische Signale

- Abschneideeffekt:
  - Betrachtet wird ein periodisches Zeitsignal

$$x(t)=\frac{1}{2}\sum_{k=-N_C}^{N_C-1} C_k e^{2\pi i k t/T}$$

mit der Periode T und der höchsten Frequenz $f_C = N_C/T$.

  - Eine Messung mit der Messdauer T_M liefert einen endlichen Abschnitt

$$x(t, T_M)=\begin{cases} x(t) & \text{für} \quad 0 \le t \le T_M \\ 0 & \text{für} \quad t<0 \vee t>T_M \end{cases}$$

4.2 Periodische Signale

- Mit dem *Rechteckfenster*

$$w_R(t, T_M) = \begin{cases} 1 & \text{für} \quad 0 \le t \le T_M \\ 0 & \text{für} \quad t < 0 \lor t > T_M \end{cases}$$

gilt:

$$x(t, T_M) = w_R(t, T_M)\, x(t)$$

- Die Fourier-Transformation des gemessenen Zeitsignals ergibt:

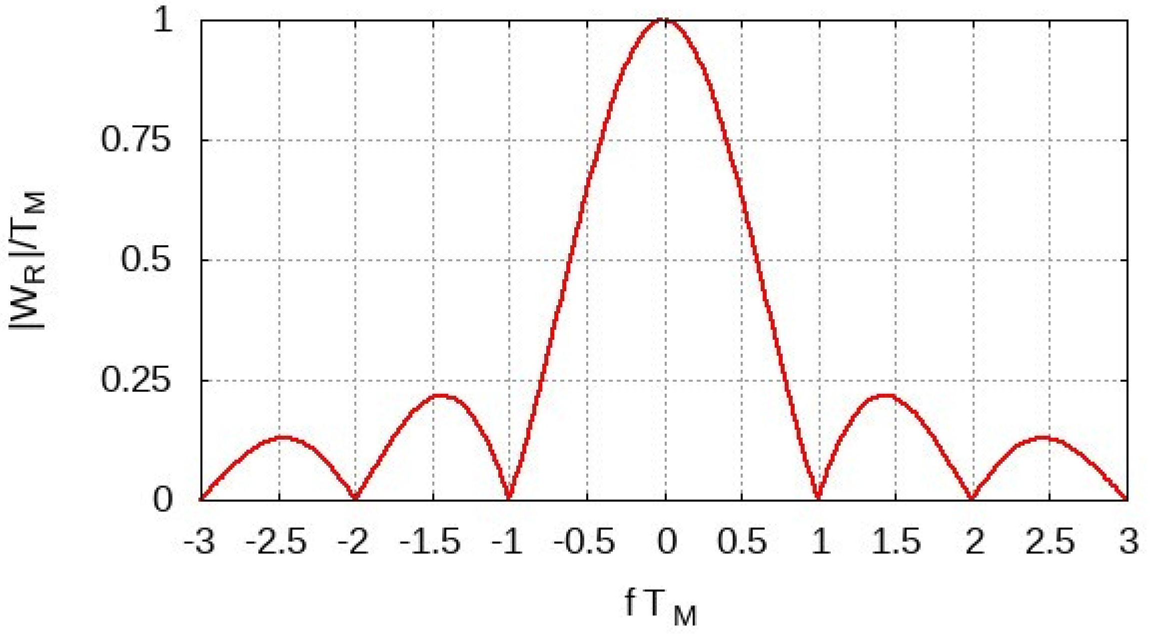
$$X(f, T_M) = \int_{-\infty}^{\infty} x(t, T_M)\, e^{-2\pi i f t}\, dt$$
$$= \frac{1}{2} \sum_{k=-N_C}^{N_C-1} C_k \int_{-\infty}^{\infty} w_R(t, T_M)\, e^{-2\pi i (f - k/T) t}\, dt$$
$$= \frac{1}{2} \sum_{k=-N_C}^{N_C-1} C_k\, W_R\left(f - \frac{k}{T}, T_M\right)$$

4.2 Periodische Signale

- Dabei ist

$$W_R(f, T_M) = \int_0^{T_M} e^{-2\pi i f t}\, dt = T_M \left(\frac{\sin(\pi f T_M)}{\pi f T_M} \right) e^{-\pi i f T_M}$$

die Fourier-Transformation des Rechteckfensters.



4.2 Periodische Signale

- Die Fourier-Transformation des Rechteckfensters hat Nullstellen bei $f T_M = n,\ n \neq 0$.
- Wenn die Frequenzen der Fourier-Reihe auf die Nullstellen von W_R fallen, tritt keine gegenseitige Beeinflussung auf:

$$\frac{k}{T} \pm \frac{n}{T_M} = \frac{l}{T} \;\rightarrow\; (k-l)\frac{T_M}{T} = \mp n \;\rightarrow\; \frac{T_M}{T} \in \mathbb{N}$$

- Wenn die Bedingung nicht erfüllt ist, liefern infolge der Seitenbänder der Fourier-Transformierten auch die benachbarten Terme einen Beitrag.
- Dieser Effekt wird als *Leakage* bezeichnet.

4.2 Periodische Signale

- Fensterfunktionen:
  - In der Praxis kann in der Regel kein ganzzahliges Verhältnis von Messdauer zu Periode gewährleistet werden.
  - Der Abschneideeffekt lässt sich verringern, wenn statt des Rechteckfensters eine andere Fensterfunktion verwendet wird, bei der die Amplitude der Seitenbänder schneller abfällt.
  - Hanning-Fenster:
    - Das Hanning- oder Hann-Fenster ist definiert durch

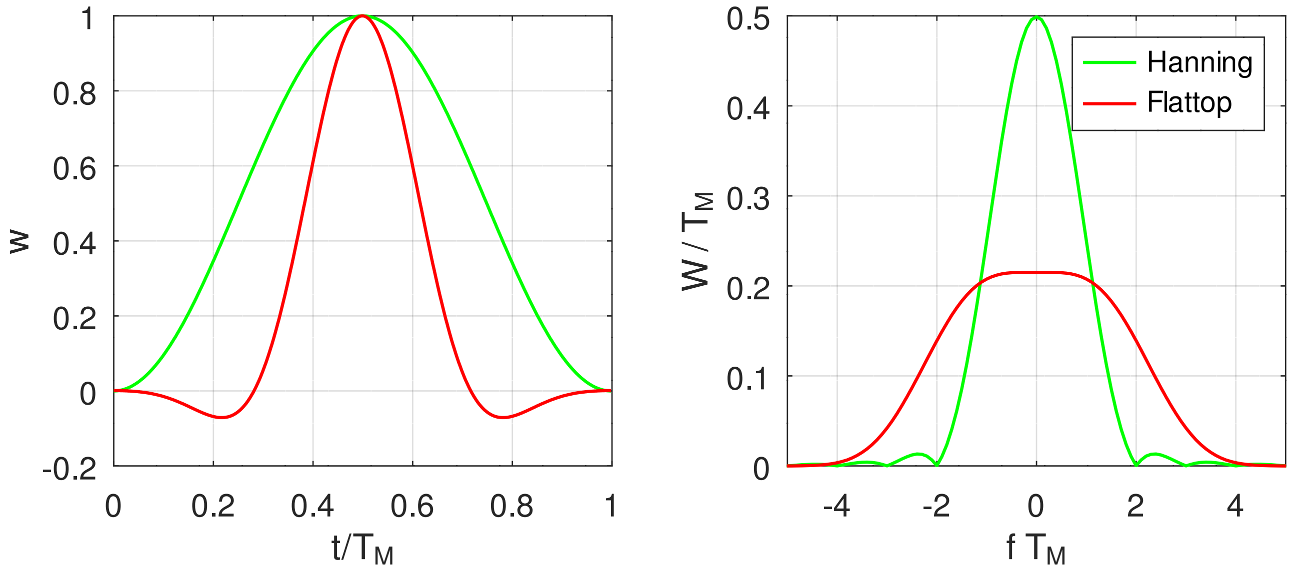
$$w_H(t, T_M) = \begin{cases} \sin^2(\pi t / T_M) & \text{für} \quad 0 \le t \le T_M \\ 0 & \text{für} \quad t < 0 \vee t > T_M \end{cases}$$

4.2 Periodische Signale

- Es liefert gute Näherungen für die Amplituden bei gleichzeitig kleiner Bandbreite und wird daher als Standard verwendet.

– Flattop-Fenster:

- Das Flattop-Fenster liefert sehr genaue Näherungen für die Amplituden bei einer größeren Bandbreite.



4.2 Periodische Signale

- Anwendung der Fensterfunktionen:
  - Die Werte der gemessenen Zeitreihe werden mit der Fensterfunktion multipliziert:

$$x_{wn} = w_n x_n$$

  - Die diskrete Fourier-Transformation der gewichteten Zeitreihe liefert die Koeffizienten X_{wk}.
  - Daraus werden Näherungen für die Amplituden berechnet:

$$|C_0| = \frac{1}{S} |X_{w0}|, \quad |C_k| = \sqrt{c_k^2 + s_k^2} = \frac{2}{S} |X_{wk}| \quad \text{mit} \quad S = \sum_{n=0}^{N-1} w_n$$

  - Für die Phasen ergeben sich keine brauchbaren Werte.
  - Die Messdauer sollte deutlich länger als eine Periode sein.

4.2 Periodische Signale

- Beispiel:
  - Untersucht wird die Zeitreihe einer periodischen Funktion mit der Periode $T = 0{,}1\ \mathrm{s}$.
  - Die Koeffizienten der Fourier-Reihe haben die Werte $s_1 = 2$, $c_3 = 5$, $s_4 = -1{,}5$, $c_4 = 2$. Alle anderen Koeffizienten sind null.
  - Die zugehörigen Frequenzen sind $f_1 = 10\ \mathrm{Hz}$, $f_3 = 30\ \mathrm{Hz}$ und $f_4 = 40\ \mathrm{Hz}$.
  - Die Messdauer beträgt $20{,}5T$.
  - Das folgende Skript untersucht den Einfluss der Fenster auf die Näherungen für die Amplituden.

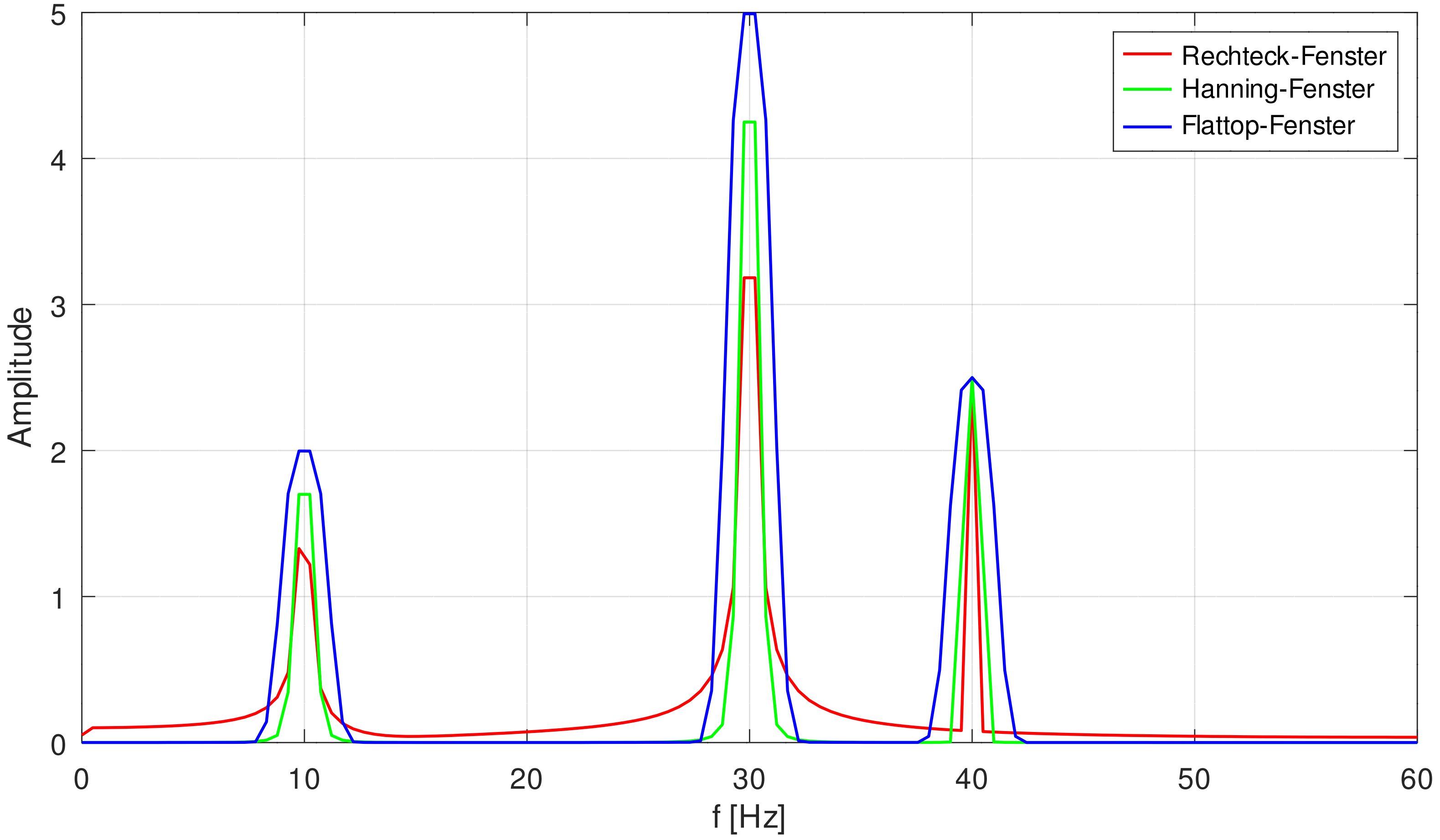
4.2 Periodische Signale

```
  pkg load signal                    % Signal-Package enthält Fenster

# Daten
  T  = 0.1;                          % Periode
  f1 = 10; s1 = 2;                   % Frequenz 1
  f3 = 30; c3 = 5;                   % Frequenz 2
  f4 = 40; s4 = -1.5; c4 = 2;        % Frequenz 3
  nT = 20.5;                         % Anzahl Perioden

# Zeitreihe
  fS = 3 * f4;                       % Abtastrate
  dt = 1 / fS;                       % Zeitschritt
  TM = nT * T;                       % Messdauer
  N  = fix(TM * fS);                 % Anzahl Messpunkte
  t  = (0 : N - 1) * dt;             % Zeitpunkte
  pt = 2 * pi * t;
  x  = s1 * sin(f1 * pt) + c3 * cos(f3 * pt);
  x += s4 * sin(f4 * pt) + c4 * cos(f4 * pt);

# Fourier-Transformation ohne Fenster
  C = 2 * fft(x) / N; C(1) = 0.5 * C(1);
```

4.2 Periodische Signale

```
  a = abs(C(1 : N/2 + 1)); f = (0 : N/2) * fS / N;

# Fourier-Transformation mit Hanning-Fenster
  wh = hanning(N)';
  Ch = 2 * fft(wh .* x) / sum(wh); Ch(1) = 0.5 * Ch(1);
  ah = abs(Ch(1 : N/2 + 1));

# Fourier-Transformation mit Flattop-Fenster
  wf = flattopwin(N)';
  Cf = 2 * fft(wf .* x) / sum(wf); Cf(1) = 0.5 * Cf(1);
  af = abs(Cf(1 : N/2 + 1));

# Vergleich
  figure(1, "position", [100, 500, 1000, 750],
            "paperposition", [0, 0, 25, 13]);
  plot(f, a, "color", "red", f, ah, "color", "green",
       f, af, "color", "blue");
  legend("kein Fenster", "Hanning-Fenster", "Flattop-Fenster");
  grid;
  xlabel("f [Hz]"); ylabel("Amplitude");
  print("v3_4_2b.svg", "-dsvg");
```

4.2 Periodische Signale



4.3 Stochastische Signale

- Der Frequenzgehalt von stationären stochastischen Prozessen wird durch das Leistungsdichtespektrum beschrieben.
- Das Standardverfahren zur Abschätzung von Leistungs- und Kreuzleistungsdichtespektren ist die Methode von Welch:
  - Ausgangspunkt ist die Gleichung

$$G_{xy}(f)=\lim_{T\to\infty}\frac{2}{T}E\left[X_k(f,T)\bar{Y}_k(f,T)\right] \quad .$$

  - Anstelle des Grenzübergangs wird eine feste Messdauer T_M verwendet, die ausreichend lang sein muss.

4.3 Stochastische Signale

- Der Erwartungswert wird durch einen über die Zeitachse gebildeten Mittelwert approximiert.
- Dazu wird die Zeitreihe in M Fenster der gleichen Dauer T_W (Länge N_W) unterteilt. Die Fenster dürfen überlappen.
- Für jedes Fenster werden die beiden Zeitreihen mit einer Fensterfunktion multipliziert. Anschließend wird die endliche Fourier-Transformation berechnet.
- Die Schätzung für das Kreuzleistungsdichtespektrum ist

$$\hat{G}_{xy}(f_k) = \frac{2}{M\,S\,T_W} \sum_{m=1}^{M} \hat{X}_{wm}(f_k, T_W)\,\bar{\hat{Y}}_{wm}(f_k, T_W), \quad f_k = k / T_W$$

mit $S = \frac{1}{N_W} \sum_n w_n^2$.

4.3 Stochastische Signale

- Das Standard-Verfahren verwendet ein Hamming-Fenster und eine Überlappung von 50 %.
- Funktionen in GNU Octave (`Signal`-Package):
  - Leistungsdichtespektrum:

    ```
    [Gxx, f] = pwelch(x,window,overlap,[],fs)
    ```

  - Kreuzleistungsdichtespektrum:

    ```
    [Gxy, f] = cpsd(x,y,window,overlap,[],fs)
    ```

  - Die Argumente **`window`** und **`overlap`** können weggelassen werden:

    ```
    [Gxx, f] = pwelch(x,[],[],[],fs)
    ```

4.3 Stochastische Signale

- Argumente:

`x, y`	Zeitreihen
`window`	Länge N_W des Fensters oder Feld mit Fensterfaktoren w_n
`overlap`	Überlappungsfaktor (zwischen 0 und 1)
`fs`	Abtastrate f_S in Hz
`Gxx, Gxy`	Leistungs- bzw. Kreuzleistungsdichte-spektrum
`f`	Frequenzen

- Für die Frequenzauflösung gilt: $\Delta f = f_S / N_W$

4.3 Stochastische Signale

- Beispiel:
  - Für die am vorderen und hinteren rechten Fahrwerksdom gemessenen Vertikalbeschleunigungen werden die Leistungs- und Kreuzleistungsdichtespektren berechnet.

```
  pkg load signal

  fname = "../../Daten/Car/az.csv";   % Eingabedatei
  lenw    = 1024;                     % Fensterlänge
  overlap = 0.5;                      % Überlappungsfaktor

# Daten einlesen und aufbereiten

  data = dlmread(fname);
  t    = data(:, 1);                  % Zeitpunkte
  dt   = mean(diff(t));               % Zeitschritt
  fs   = 1 / dt;                      % Abtastrate
  az   = data(:, [3, 5]);             % Beschl. vorne / hinten
```

4.3 Stochastische Signale

```
# Leistungs - und Kreuzleistungsdichtespektren

  [Gvv, f] = pwelch(az(:, 1), lenw, overlap, [], fs);
  [Ghh, f] = pwelch(az(:, 2), lenw, overlap, [], fs);
  [Ghv, f] = cpsd(az(:, 2), az(:, 1), lenw, overlap, [], fs);

# Kohärenz

  ghv = Ghv .* conj(Ghv) ./ (Gvv .* Ghh);

# Ausgabe

  figure(1, "position", [1100, 400, 1000, 500],
            "paperposition", [0, 0, 25, 13.5]);

  subplot(1, 2, 1);
     semilogy(f(2 : end), Gvv(2 : end), "color", "green",
              f(2 : end), Ghh(2 : end), "color", "red");
     legend("G_{vv}", "G_{hh}");
     grid;
     xlabel("f [Hz]");
     ylabel("[m^2/(s^4 Hz)]")
```

4.3 Stochastische Signale

```
subplot(1, 2, 2);
   plot(f(2: end), ghv(2: end), "color", "blue");
   legend('\gamma^2_{hv}');
   grid;
   xlabel("f [Hz]");

print("v3_4_3.svg", "-dsvg");
```

4.3 Stochastische Signale