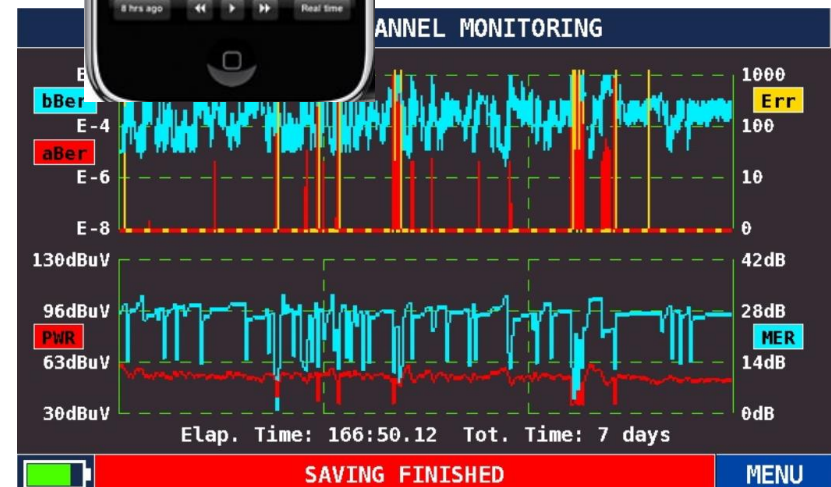
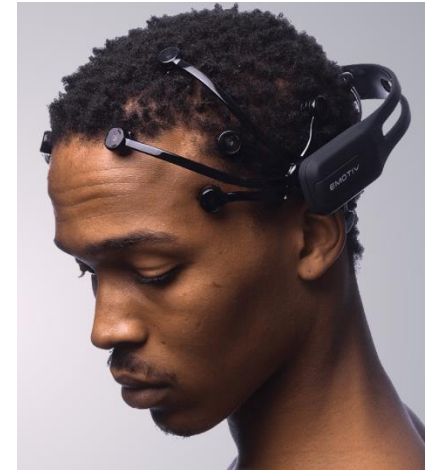
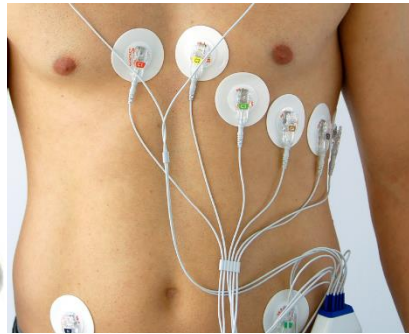


Signalverarbeitung

Prof. Dr.-Ing. Andreas Meisel





Inhaltsübersicht

A) Einführung

- Was ist Signalverarbeitung
- Mathematische Grundlagen und historische Annäherung

B) Theorie der Signalverarbeitung

- fundierte Herleitung der systemtheoretischen Grundlagen
- Konsequenzen für Signalverarbeitungsanwendungen

C) Praktische digitale Signalverarbeitung

- Zeitdiskrete Signale und Systeme
- Entwurf von Digitalfiltern
- Frequenz- Analysetechniken
- ausgewählte Anwendungen

.....



Vorgehensweise

Vortragsfolien → Pub

Vertiefungen → Surface/Tafel (+ Pub oder owncloud)

Übungsaufgaben → Surface/Tafel (+ Pub oder owncloud)

Übungsblätter zum Selberrechnen → Pub oder owncloud

Übungs-/Praktikumsanleitungen → Pub oder owncloud

Public-Verzeichnis der Informatik

- <https://users.informatik.haw-hamburg.de/~meisel/pub/>

Owncloud

<https://owncloud.informatik.haw-hamburg.de/index.php/s/1muXfmiVv541MCB>

PW: MSL_LV



Literatur und Onlinequellen

Grundlagenwerke

Digitale Signalverarbeitung, Daniel Ch. von Grünigen,
Fachbuchverlag Leipzig

Signalverarbeitung mit MATLAB und Simulink, Hoffmann/Quint
Oldenbourg Verlag

Signalverarbeitung – Analoge und dig. Signale, Systeme und Filter, Martin Werner,
Springer Vieweg ([online im Hochschulnetz verfügbar](#))

Digitale Signalverarbeitung mit MATLAB, Martin Werner,
Vieweg+Teubner ([online im Hochschulnetz verfügbar](#))

Digitale Signalverarbeitung 1 und 2, Hans W. Schüßler,
Springer ([online im Hochschulnetz verfügbar](#))

Systemtheorie online

<http://www.eit.hs-karlsruhe.de/mesysto/quicklink/startseite.html>

uvm.



Literatur

Weitergehende Literatur

Digitale Signalverarbeitung – Bausteine, Systeme, Anwendungen,
Daniel Ch. von Grünigen, Print & Media

DAFX – Digital audio Effects, Udo Zölzer,
Wiley Interscience



Weiterführende (Online-) Publikationen + Recherche

IEEE Xplore (aus Hochschulnetz)

ACM Digital Library (aus Hochschulnetz)

Springer: International Journal of Computer Vision (aus Hochschulnetz)

Citeseer (frei)

Google Scholar (frei)



Eingesetzte Softwarewerkzeuge / Bibliotheken

Vorlesung:

MAPLE 14 (Campuslizenz)

Matlab (Campuslizenz)

Übung:

Matlab (Campuslizenz)



A

Einführung



A

1. Was ist Signalverarbeitung ?

- **Signale**
- Systeme
- Signalverarbeitung



1 Eingrenzung

Signal = zeitlicher Verlauf einer beobachteten, physikalischen Größe

- kann skalar, vektoriell oder komplex sein
- kann wert- und zeitkontinuierlich und/oder wert- und zeitdiskret sein

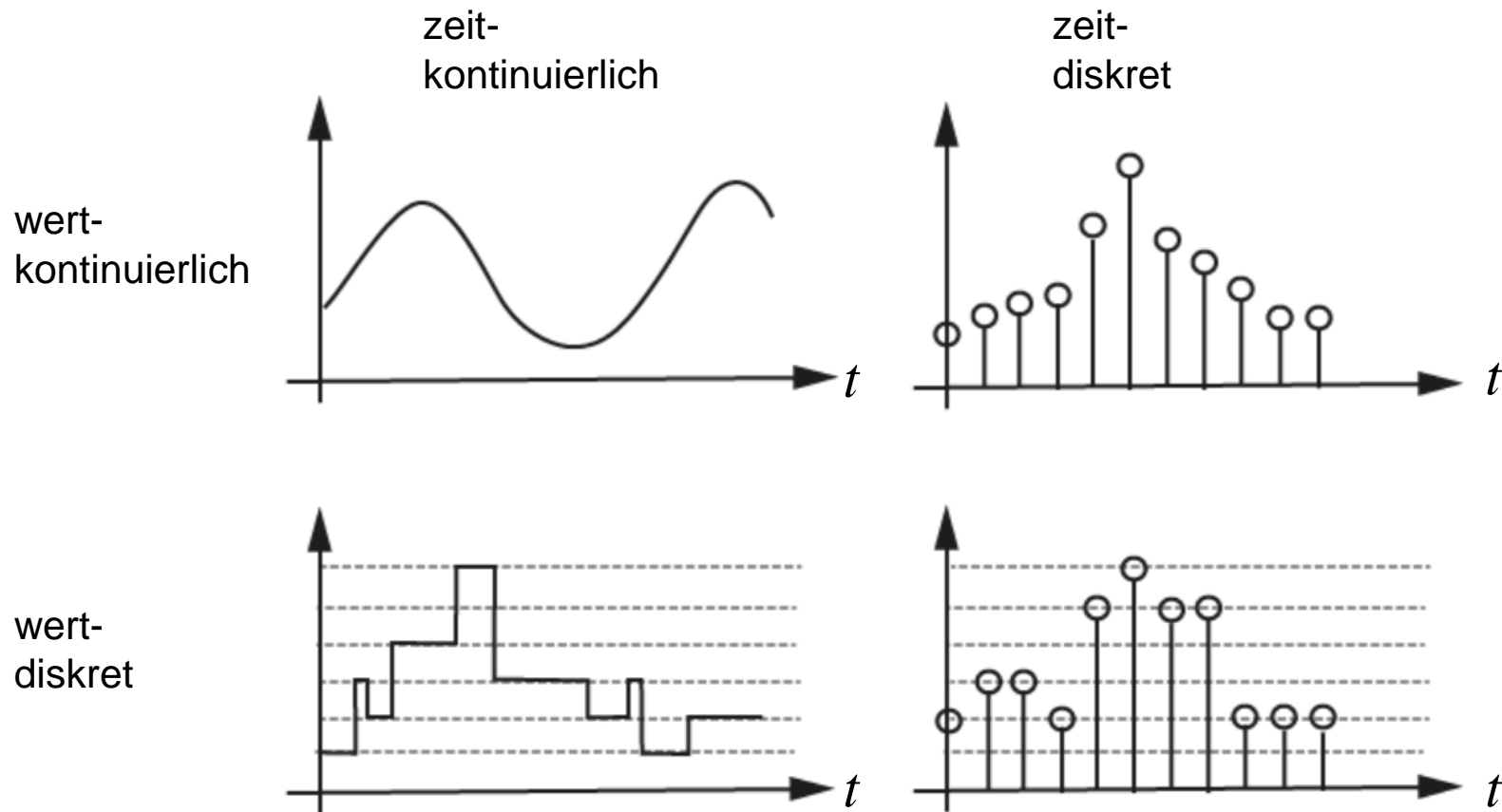
Beispiele:

- Sprache, Musik, Geräusche,
- Temperaturverlauf, Spannungsverlauf,
- Koordinaten einer räumlichen Bewegung,
- Börsenkurse,
- medizinische Signale (z.B. EKG, EEG)

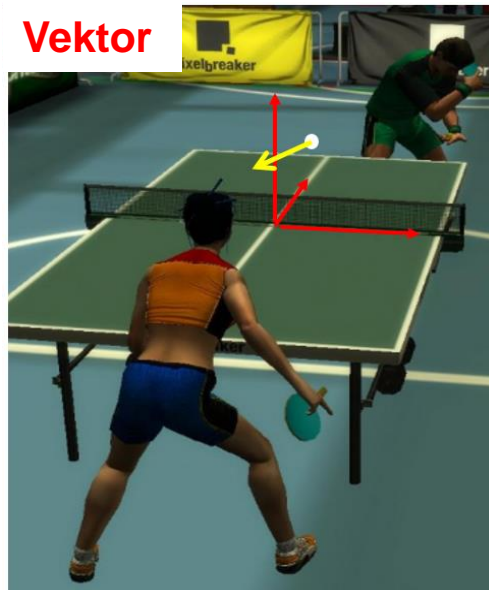
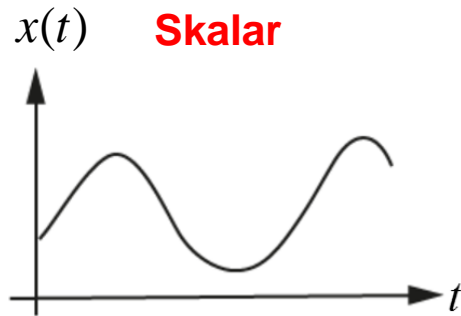


2 Signalklassen

2.1 Kontinuierliche und diskrete Signale

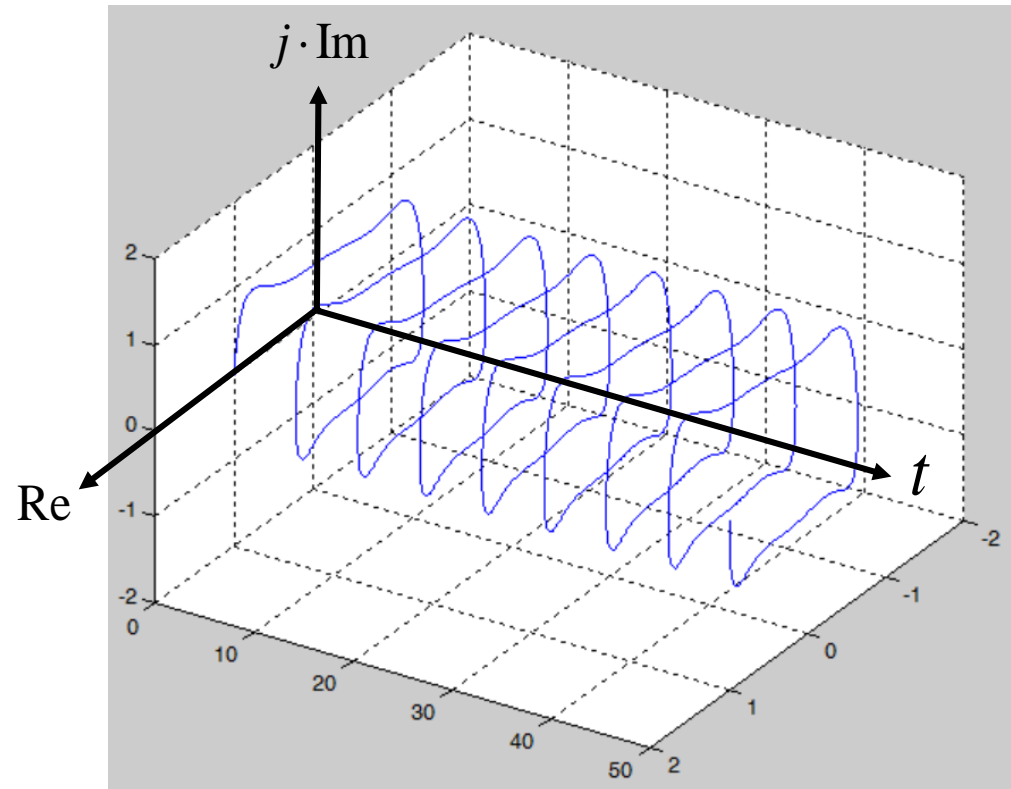


2.2 Skalare, vektorielle und komplexe Signale



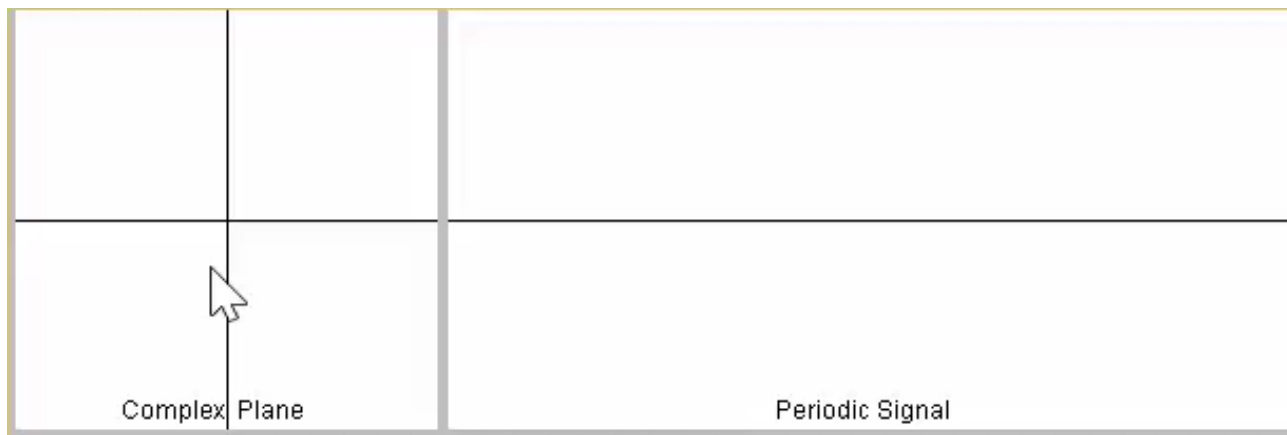
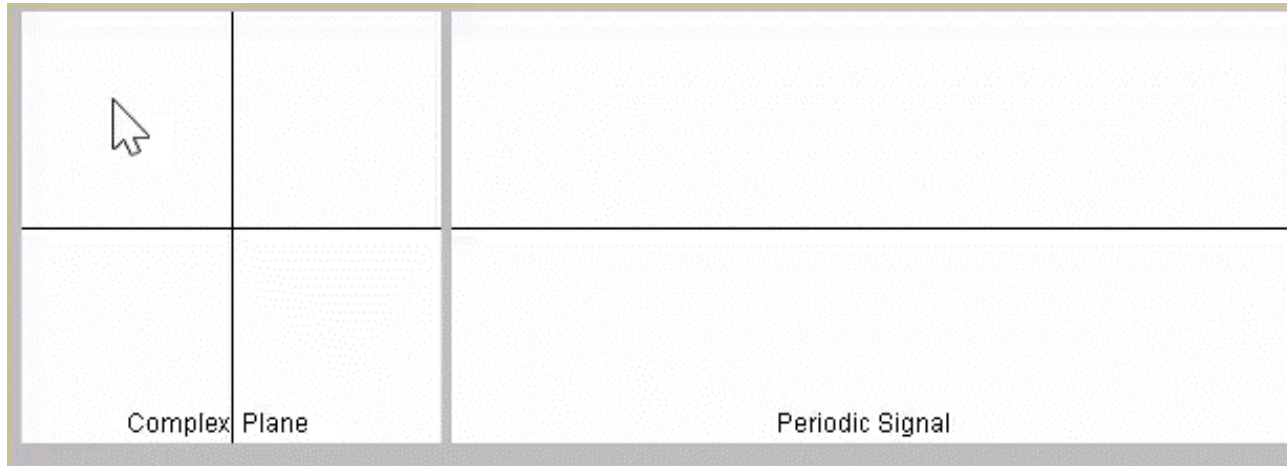
$$\vec{x}(t) = [x(t), y(t), z(t)]^T$$

Komplex





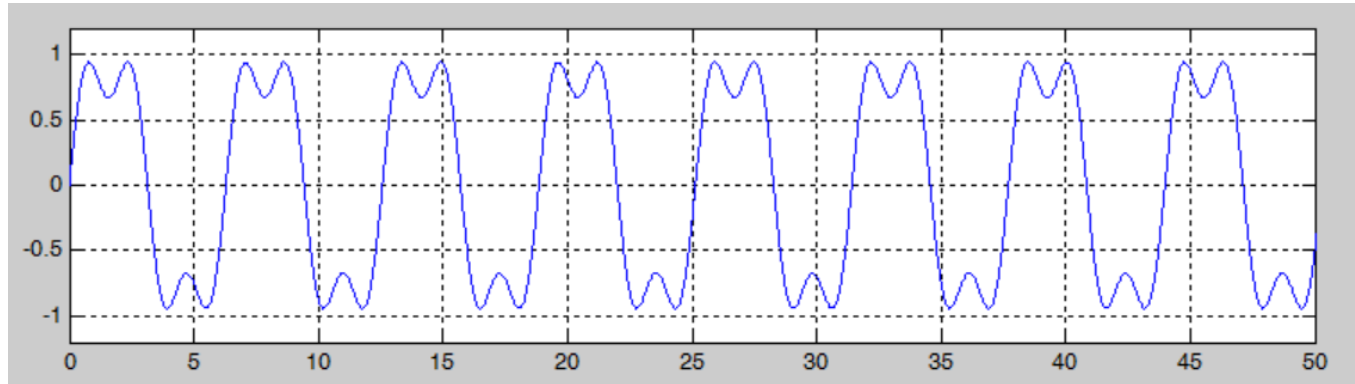
Beispiel: Komplexwertige Funktionen $\underline{x}(t) = x(t) + j \cdot y(t) = r(t) \cdot e^{j\varphi(t)}$



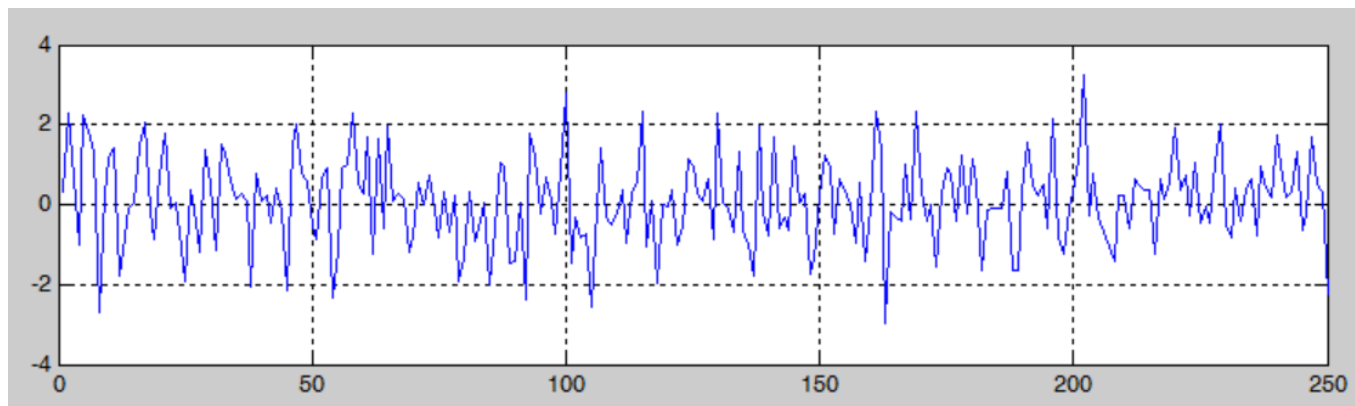


2.3 *Determinierte und stochastische Signale*

Determiniert = durch Funktionen oder Rechengvorschrift beschreibbar



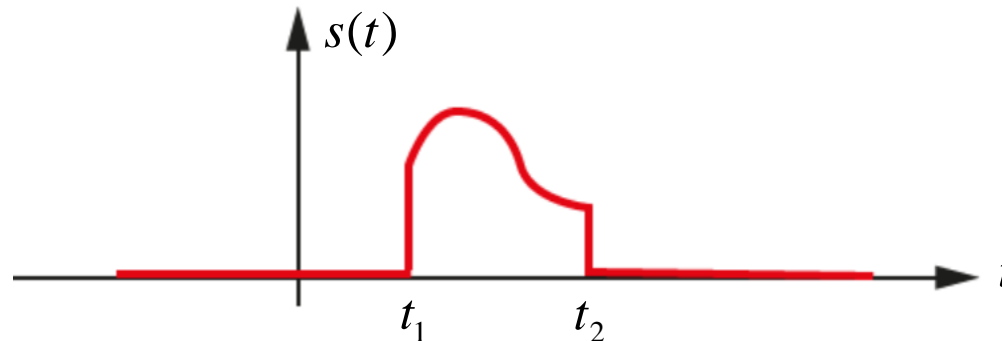
Stochastisch = nur statistisch beschreibbar (Rauschen, Sprache, Datenströme, ...)



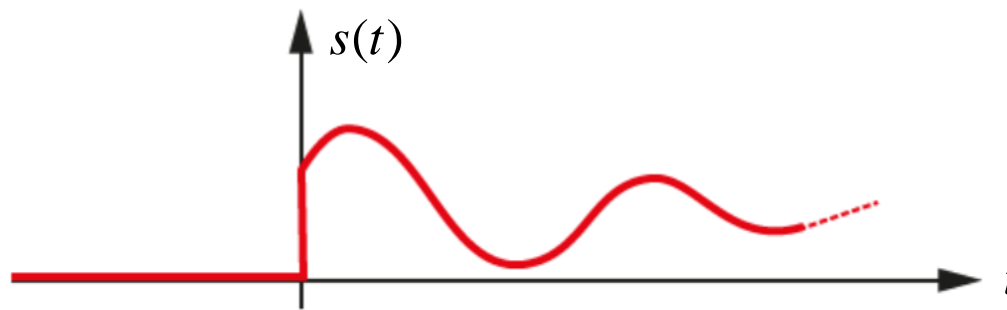


2.5 Zeitlich begrenzte Signale

beidseitig begrenzt = $s(t) \neq 0$ für $t_1 \leq t \leq t_2$, sonst $s(t) = 0$

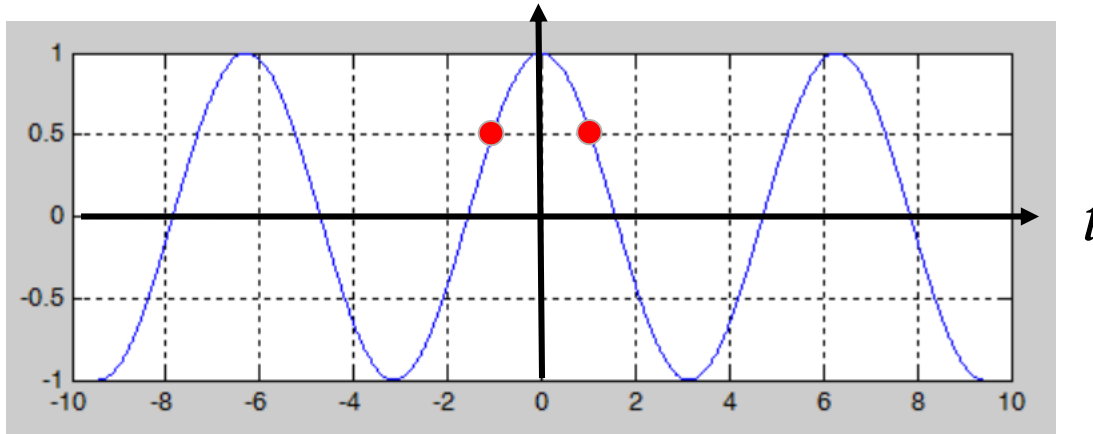


Kausal (linksseitig begrenzt) $s(t) \neq 0$ für $t \geq 0$, sonst $s(t) = 0$

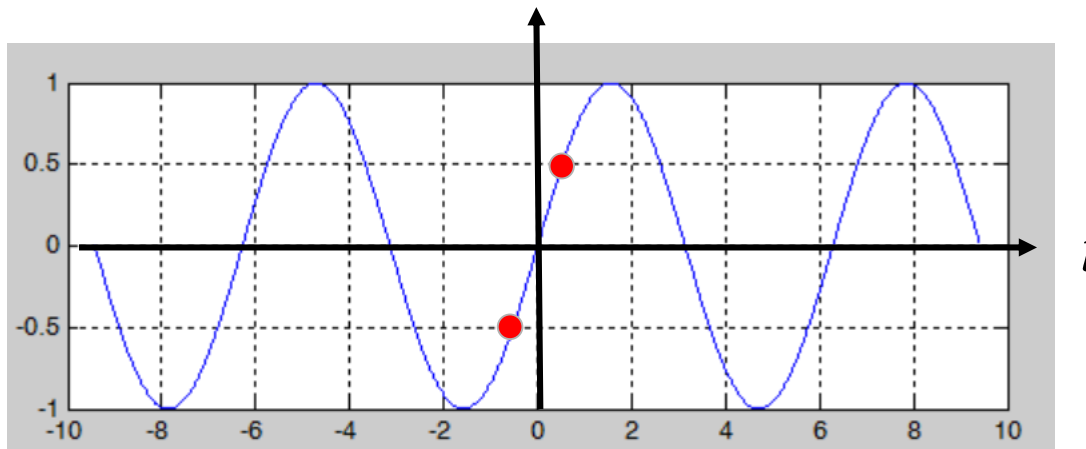


2.6 Symmetrieeigenschaften von Signalen

gerade Funktion = $s(t) = s(-t)$



ungerade Funktion = $-s(t) = s(-t)$





unsymmetrische Signale

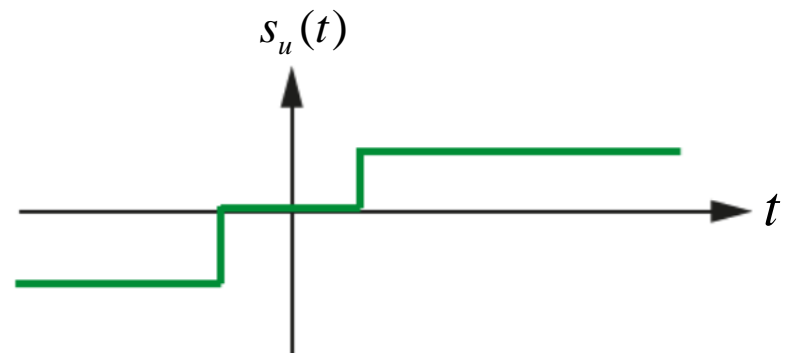
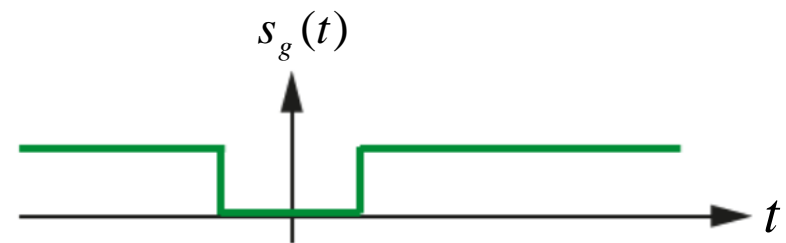
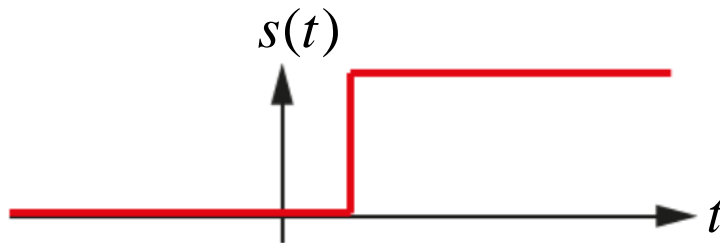
Unsymmetrische Signale lassen sich durch eine Summe einer geraden mit einer ungeraden Funktion darstellen.

$$s(t) = s_g(t) + s_u(t)$$

Der gerade und ungerade Anteil wird bestimmt mit

$$s_g(t) = \frac{1}{2} \cdot [s(t) + s(-t)]$$

$$s_u(t) = \frac{1}{2} \cdot [s(t) - s(-t)]$$

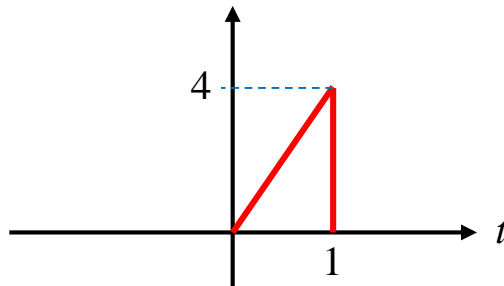




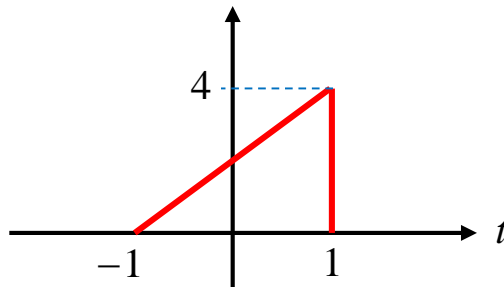
Übung: Gerade und ungerade Signale

Skizzieren Sie zu folgenden Signalen den geraden und ungeraden Signalanteil unter Angabe von Kennwerten.

a)



b)





A

1. Was ist Signalverarbeitung ?

- Signale
- **Systeme**
- Signalverarbeitung



1 Einführung

System = Einrichtung, die auf ein Eingangssignal mit einem Ausgangssignal antwortet. Systeme können sein

- linear oder nichtlinear
- kontinuierlich (analog) oder zeitdiskret (digital)
- stabil oder instabil
- kausal oder nichtkausal

Beispiele:

- Tiefpass, Hochpass, Bandpass, Bandsperre
- Audioeffekt (Phaser, Flanger, Chorus, Dynamikkompressor,)
- Gitarrenamp-Simulation
- schwingendes mechanisches System (Feder-Masse-Schwinger, ...)
- Lautsprecher, Raumakustik
- Störungsfilter (EKG oder EEG)
- Signalglättungsfilter
-



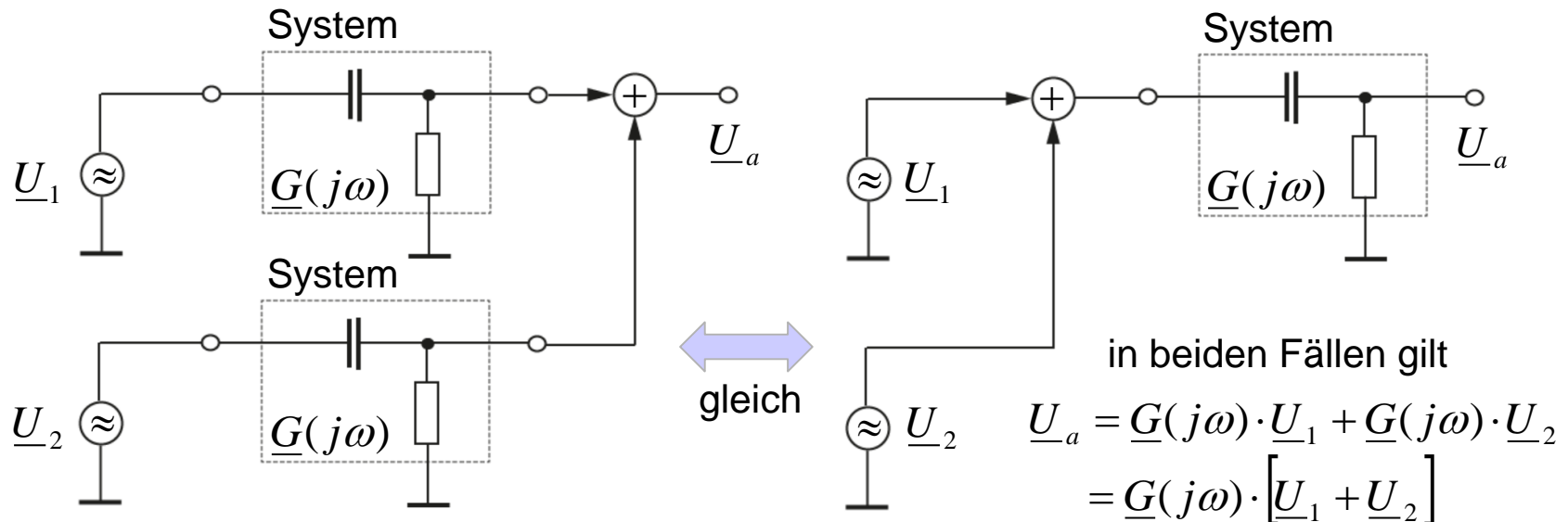
2 Systemklassen

2.1 Lineare und nichtlineare Systeme

Bei linearen Systemen ist das Ergebnis unabhängig davon, ob

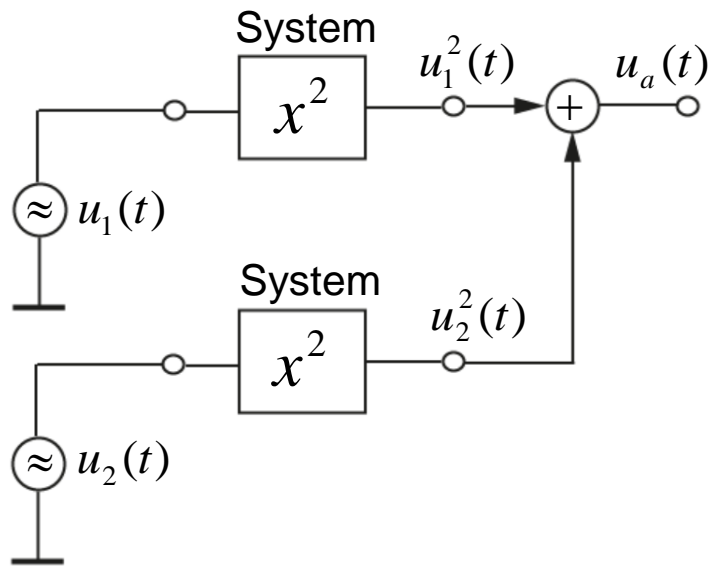
- mehrere Signale unabhängig voneinander das System durchlaufen und dann die Ausgangssignale addiert werden *oder*
- die gleichen Signale erst addiert werden und das Summensignal dann das System durchläuft.

Beispiel: lineares System (z.B. hier RC-Hochpass)

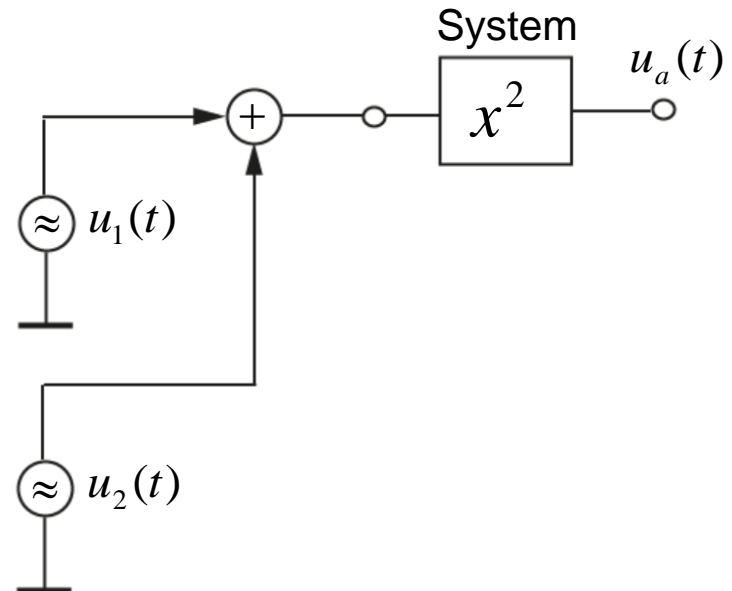




Beispiel: nichtlineares System



$$u_a = u_1^2 + u_2^2$$



$$u_a = (u_1 + u_2)^2 = u_1^2 + 2u_1 u_2 + u_2^2$$

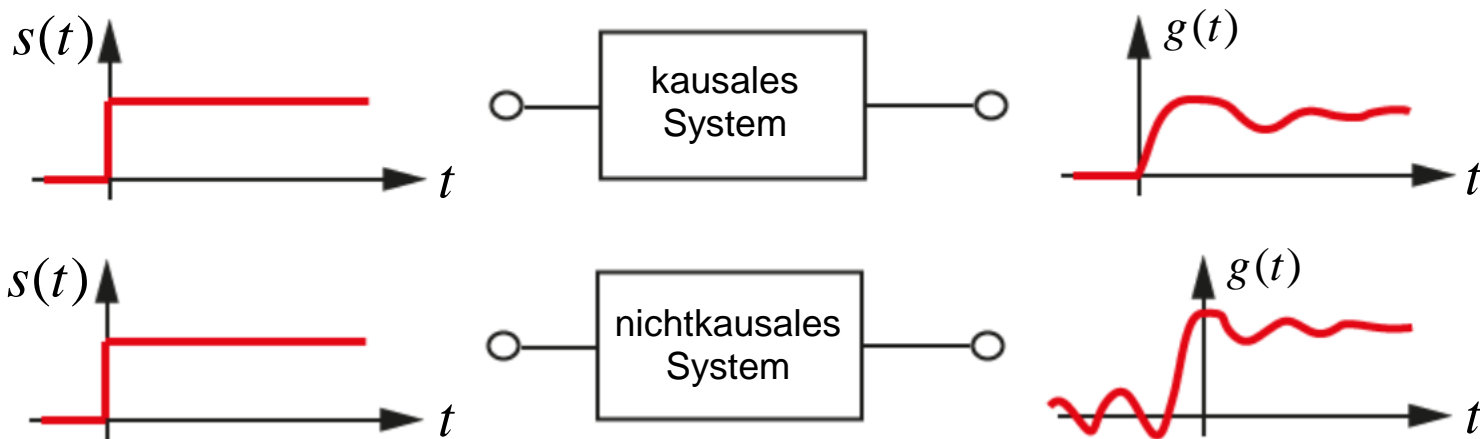
2.2 Kausale und nichtkausale Systeme

Ein System ist nur dann kausal, wenn das Ausgangssignal ausschließlich eine Wirkung des Eingangssignals ist.

Ein System bei dem die Ausgangsgröße vor der Eingangsgröße auftritt, ist nichtkausal.

Reale Systeme sind immer kausal.

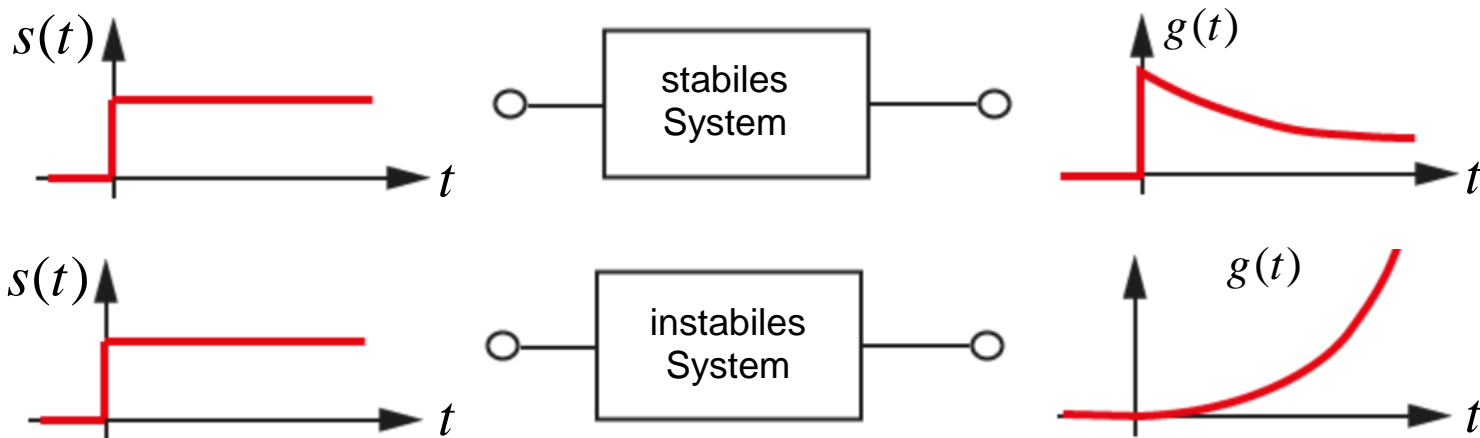
Bei theoretischen Überlegungen sind aber auch nichtkausale Systeme möglich.



2.3 Stabile und instabile Systeme

Ein System ist dann stabil (Bibo-Stabilität), wenn das Ausgangssignal bei beschränktem Eingangssignal nicht über alle Grenzen anwächst.

bibo = bounded input, bounded output





2.4 Zeitinvariante und zeitvariante Systeme

Systeme, deren Reaktion auf ein Eingangssignal unabhängig vom Zeitpunkt immer gleich ist, sind *zeitinvariant*.

Beispiel: Zeitinvariantes System

Ein RC-Tiefpass verhält sich zu jedem beliebigen Zeitpunkt immer gleich (wenn man von der Alterung der Bauteile absieht).

Beispiel: Zeitvariantes System

Die Wirkung der Steuerbewegung einer Rakete wird zu verschiedenen Zeitpunkten unterschiedlich ausfallen, da die Masse und damit die Trägheit der Rakete ständig abnimmt.



A

1. Was ist Signalverarbeitung ?

- Signale
- Systeme
- **Signalverarbeitung**



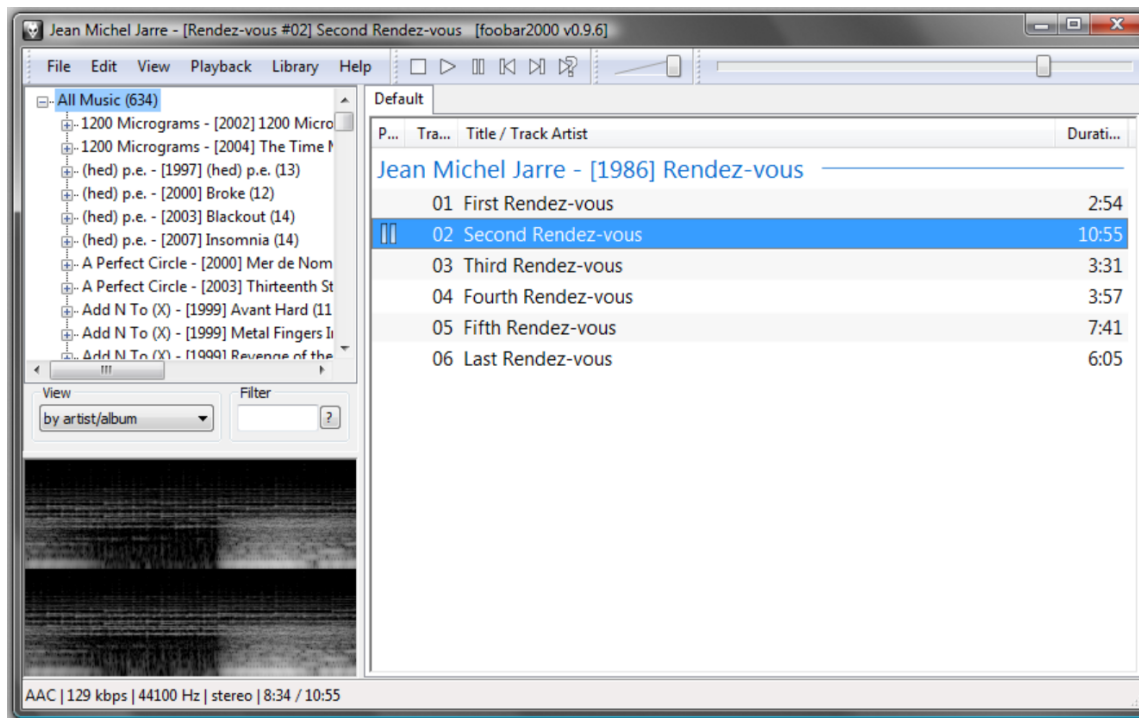
1 Aufgabenstellungen

- Änderung der Frequenzzusammensetzung (Hochpass, Tiefpass, Equalizer)
- Signalglättung (gleitender Mittelwert, Median, ...)
- Änderungen des Abtasttaktes (Interpolation und Dezimation)
- Arithm./log. Operationen mit Signalen (Signaladdierer, Signalmultiplizierer, ...)
- Signalformung (Waveshaper, Distortion-Effekt, ...)
- Effektdesign (Echo, Hall, Chorus, Dynamikkompressor,)
- Frequenztransformationen (Pitchshifter, Modulation)
- Extraktion der Einhüllenden (Demodulation,)
- Signalvergleich (Korrelation, dynamic-time-warping, ...)
- Merkmalsextraktion (Takt der Musik, Musikgenre,)
- Signalprädiktion, Systemidentifikation, ...
- Modulation (im Sender) und Demodulation (beim Empfänger)
- Spracherkennung, Sprechererkennung, Instrumentenerkennung, ...
- Audio to MIDI, Guitar to MIDI, ...
- uvm.

2 Beispiele

2.1 Beispiel: Audioplayer und Audioverarbeitung

Foobar 2000 - Musicplayer



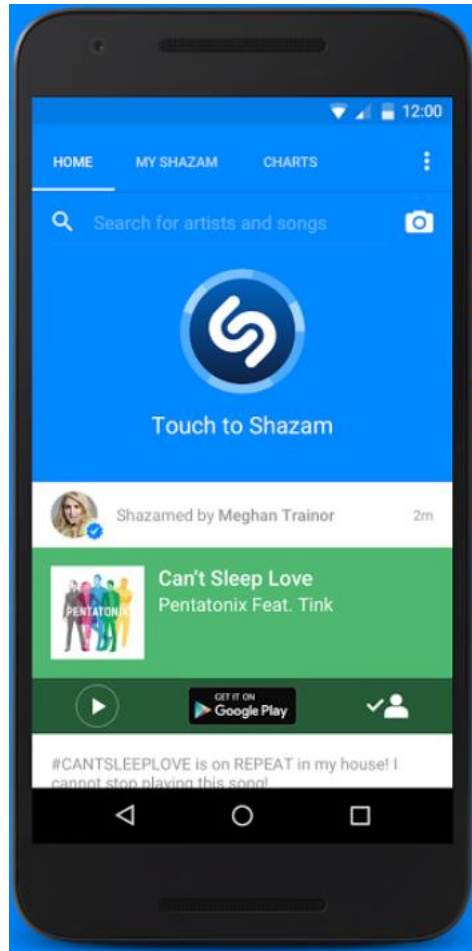
- Musik abspielen
- Frequenzanalyse
- Klangregelung
- Levelizer
-

Studiosoftware mit VST-Plugins (z.B. Ableton Live)



Tablet/Smartphone Musikapplikationen

Shazam - Musikererkennung



Nave – Wavetable-Synthesizer





2.2 Beispiel: Sensorsignalverarbeitung



BRAINFLIGHT

Gehirn lenkt Drohne

"Ein bisschen weiter nach oben", denkt sich der Pilot und schon schwebt die **Drohne** höher. Das ist keine Szene aus einem Science-Fiction-Film, sondern das Ziel des Projekts Brainflight. Es könnte eines Tages auch zur Steuerung von Passagierflugzeugen genutzt werden.



EDSAP

Samsung entwickelt Schlaganfall-Früherkennungssystem

Einige Sensoren und ein **Smartphone** sollen ausreichen, um das Risiko eines Schlaganfalls vorab zu erkennen. Entwickelt wurde das Früherkennungssystem von **Samsung**-Mitarbeitern, die sich sonst mit ganz anderen Geräten befassen.



BCI

Roboter wird per Gedanken gesteuert

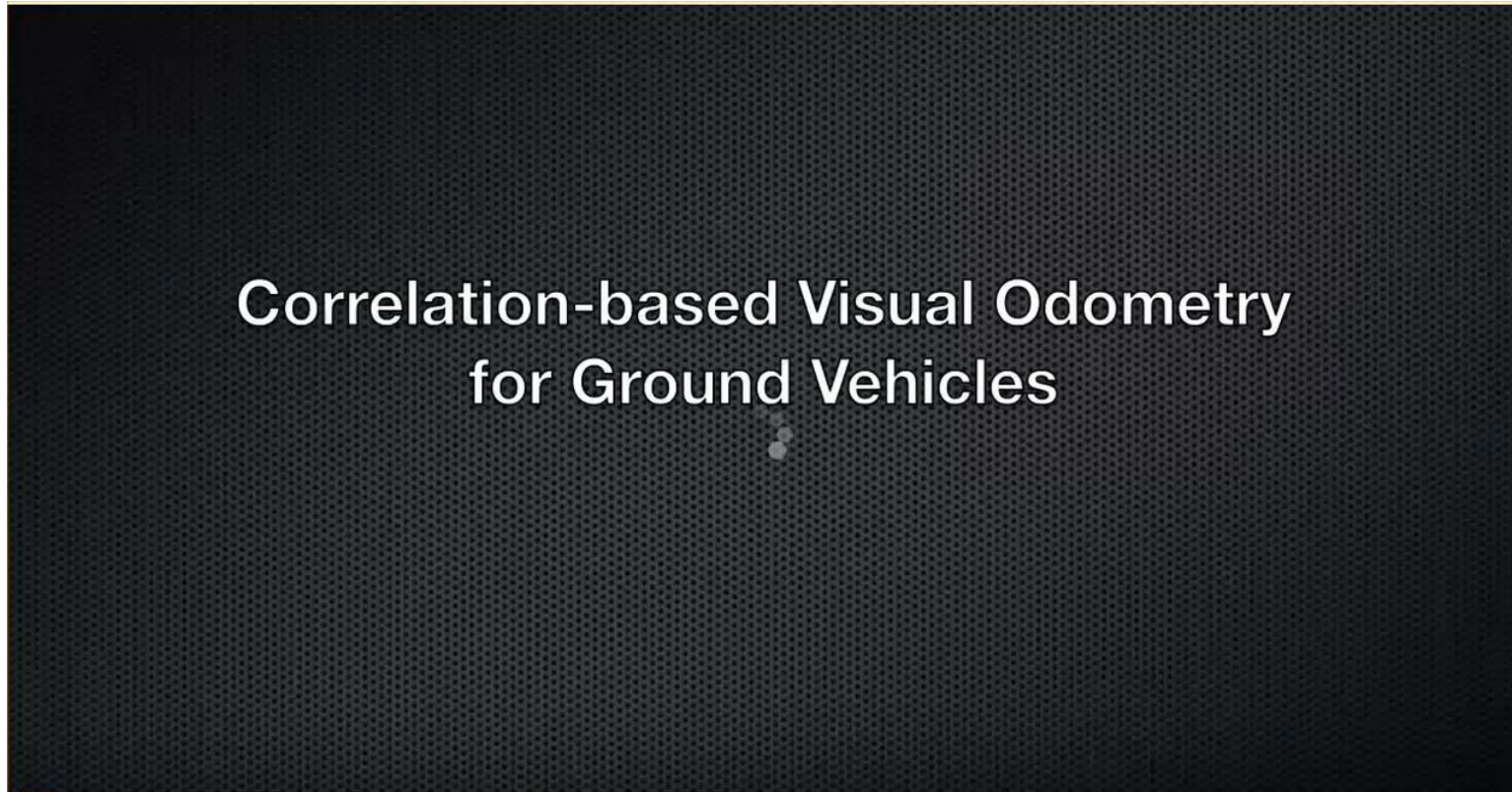
Ein französisch-japanisches Forscherteam hat eine Gedankensteuerung für einen Roboter entwickelt. Der Proband sieht optische Reize auf einem Bildschirm, die in Steuersignale umgesetzt werden.

Gesehen unter

<http://www.golem.de/specials/bci/>



2.3 Beispiel: Odometrie mit Hilfe von Korrelation

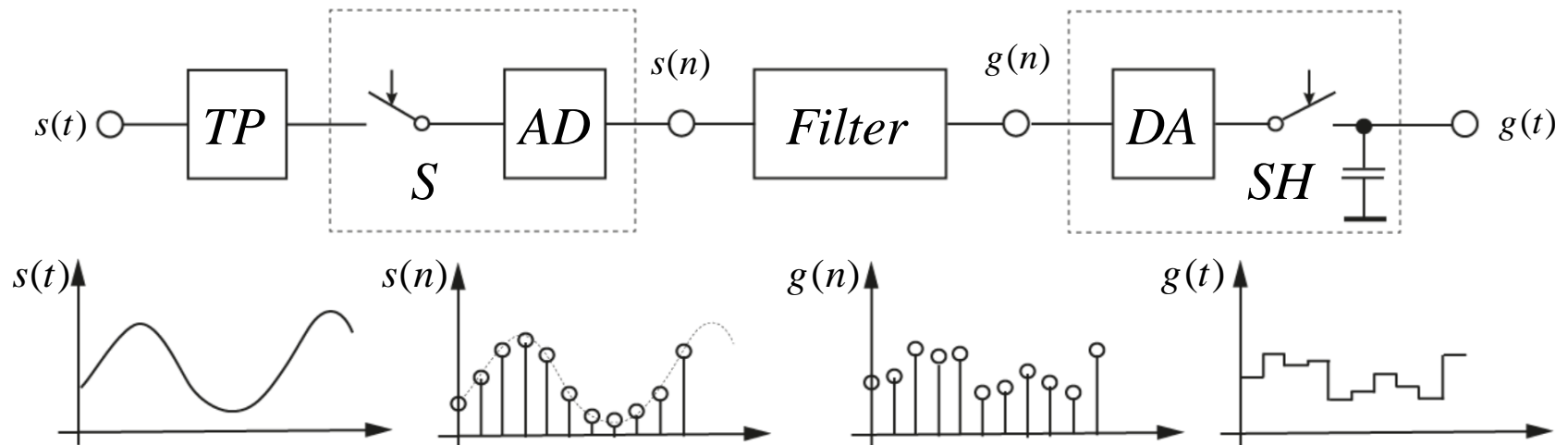


https://www.youtube.com/watch?v=Qjh3Fgw_10g

Navid Nourani

3 Struktur von digitalen Signalverarbeitungsanwendungen

3.1 Grundstruktur



TP Tiefpass

AD Analog-Digital-Wandler

DA Digital-Analog-Wandler

S Sampler

S Sample-And-Hold



3.2 Zielhardware

Signalverarbeitungsanwendungen haben in den meisten Fällen (z.B. bei Realtime-Anwendungen) einen sehr hohen Rechenleistungsbedarf.

Beispiel: Audioanwendungen haben eine Taktfrequenz von 44.1 kHz (Audio-CD) oder auch 48 kHz, 96 kHz oder 192 kHz bei der Audibearbeitung.

Je nach Anwendungsfall läuft die Signalverarbeitungsanwendung auf

- digitalen Signalprozessoren,
- kundenspezifischer Hardware, z.B. ASIC's,
- auf programmierbarer Hardware, z.B. FPGA's ,
- auf Microcontrollern (ARM etc.),
- auf PCs und Notebooks,
- auf Tablets und Smartphones.



s. nächste Seite



ASIC – Application-specific integrated Circuit

- hohe Einmalkosten
- geringe Stückkosten → Massenprodukte
- Anfertigung durch Spezialfirmen nach Kundenspezifikation
- hohe Rechenleistung für Spezialaufgaben durch inhärente Parallelität
- hohe Rechenleistung pro Chipfläche, geringer Stromverbrauch
- einmal fertiggestellt ist der ASIC nicht mehr veränderbar (unflexibel)

FPGA – Field programmable gate array

- viel geringere Entwicklungskosten als ASICs
- bei hohen Stückzahlen teurer als ASICs, bei mittleren Stückzahlen günstiger
- Anfertigung inhouse (sofern sich eine FPGA-Entwicklungsabteilung für die Firma lohnt) oder durch Spezialfirmen nach Kundenspezifikation
- hohe Rechenleistung für Spezialaufgaben durch inhärente Parallelität (< als ASIC)
- hohe Rechenleistung pro Chipfläche (< ASIC), geringer Stromverbrauch (> ASIC)
- flexibel veränderbar (updatefähig)



Microcontroller

- konventionelle Prozessorarchitektur (Prozessor, Speicher, Peripherie, Befehlssatz)
Beispiel: ARM Cortex – Programmierung typ. C/C++
- viel geringere Entwicklungskosten als FPGA's
- typ. inhouse Entwicklung
- sequentielle Aufgabenbearbeitung, daher geringere Rechenleistung als FPGA's
- für alle Zwecke einsetzbar (general purpose), Komplexität der Aufgabenstellung nur durch Speicher begrenzt
- flexibel veränderbar (updatefähig)



4 Verwandte und angrenzende Themengebiete

- Bildverarbeitung
- (dig.) Regelungstechnik
- Simulation zeitkontinuierlicher Vorgänge
- spezielle Neuronale Netze (z.B. Time-Recurrent-Netze)
- ...



A

2. Mathematische Grundlagen und Historie

- **Komplexe Zahlen** – eine kurze Wdh.
- Fourierreihe
- Diskrete Form der Fourierreihe



1 Imaginäre Einheit und anschauliche Deutung komplexer Zahlen

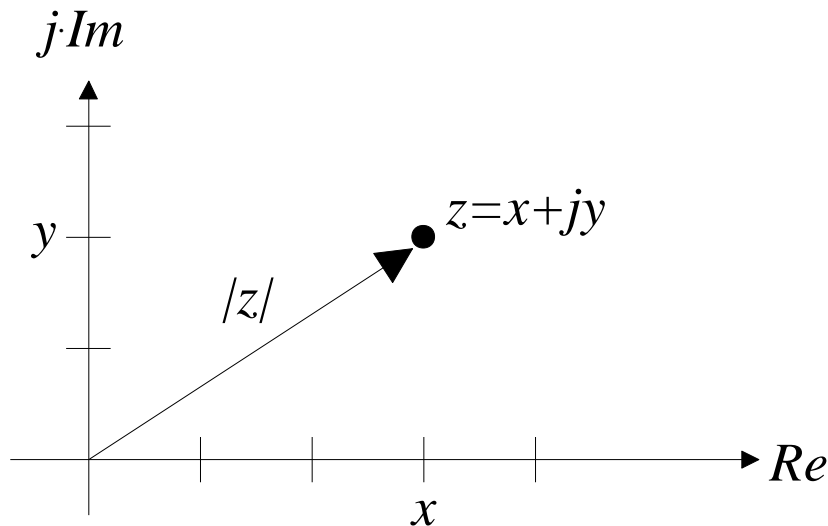
Eine komplexe Zahl $z = x + jy$ markiert ein Punkt in der komplexen Zahlenebene.

x : Realteil der komplexen Zahl

y : Imaginärteil der komplexen Zahl

Die besonderen Eigenschaften komplexer Zahlen (das was sie von einem Vektor unterscheidet) kommt von einer besonderen Eigenschaft der imaginären Einheit j , nämlich das gilt:

$$j^2 = -1$$

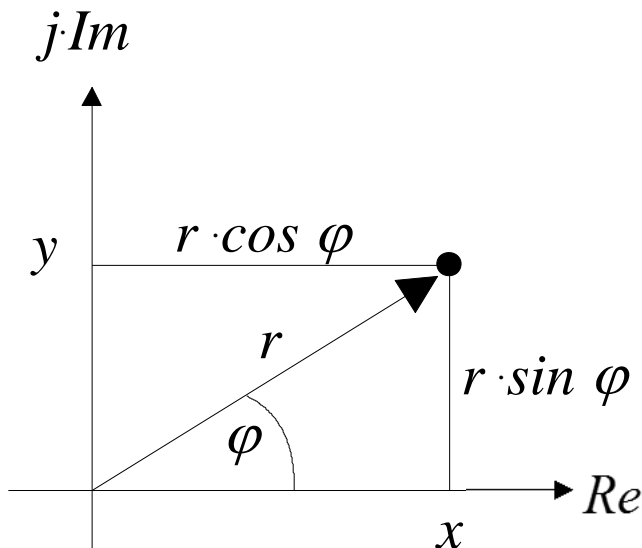




2 Darstellung komplexer Zahlen in Polarkoordinaten

Anstelle der kartesischen Darstellung $z=x+jy$ kann der Punkt auch durch ein Zeiger der Länge r und den Winkel φ markiert werden.

Damit gilt: $z = x + jy = r \cos \varphi + j \cdot r \sin \varphi$



kartesisch \rightarrow polar

$$r = \sqrt{x^2 + y^2}$$

$$\varphi = \arctan\left(\frac{y}{x}\right)$$

polar \rightarrow kart.

$$\sin(\varphi) = \frac{y}{r} \Leftrightarrow y = r \cdot \sin(\varphi)$$

$$\cos(\varphi) = \frac{x}{r} \Leftrightarrow x = r \cdot \cos(\varphi)$$



3 Euler-Darstellung komplexer Zahlen

Leonhard Euler entdeckte einen (überraschenden) Zusammenhang zwischen der Exponentialfunktion und den trigonometrischen Funktionen.



$$r \cdot (\cos \varphi + j \sin \varphi) = r \cdot e^{j\varphi}$$



- kompakte Schreibweise für kompl. Zahlen
- neue (einfache) Rechenregeln

Beispiel: $\underline{z} = 3 + j4 \rightarrow r = \sqrt{3^2 + 4^2} = 5, \quad \varphi = \arctan\left(\frac{4}{3}\right) = 53.13^\circ$

$$\underline{z} = 3 + j4 = 5 \cdot [\cos(53.13^\circ) + j \sin(53.13^\circ)] = 5 \cdot e^{j \cdot 53.13^\circ}$$

$\underbrace{\hspace{10em}}$
Kartesische Form

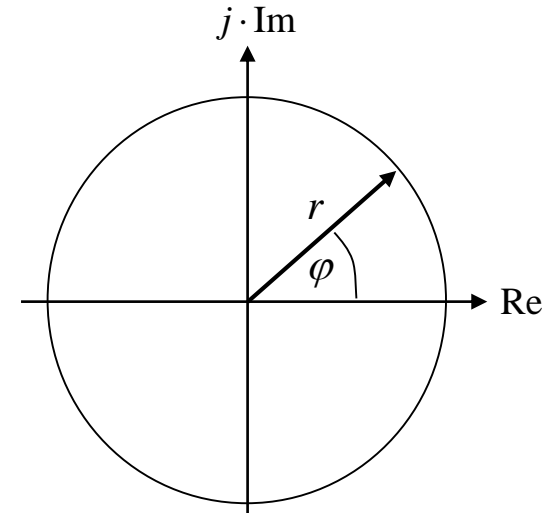
$\underbrace{\hspace{10em}}$
Polarkoordinatenform

$\underbrace{\hspace{10em}}$
Exponentialform

Anm.: Dieser Zusammenhang folgt aus der Reihenentwicklung beider Funktionen (o.Bew.).

4 Drehzeiger (Phasor)

$r \cdot e^{j\varphi}$ ist ein Zeiger der Länge r
der unter dem Winkel φ auf einen
Punkt der komplexen Ebene zeigt



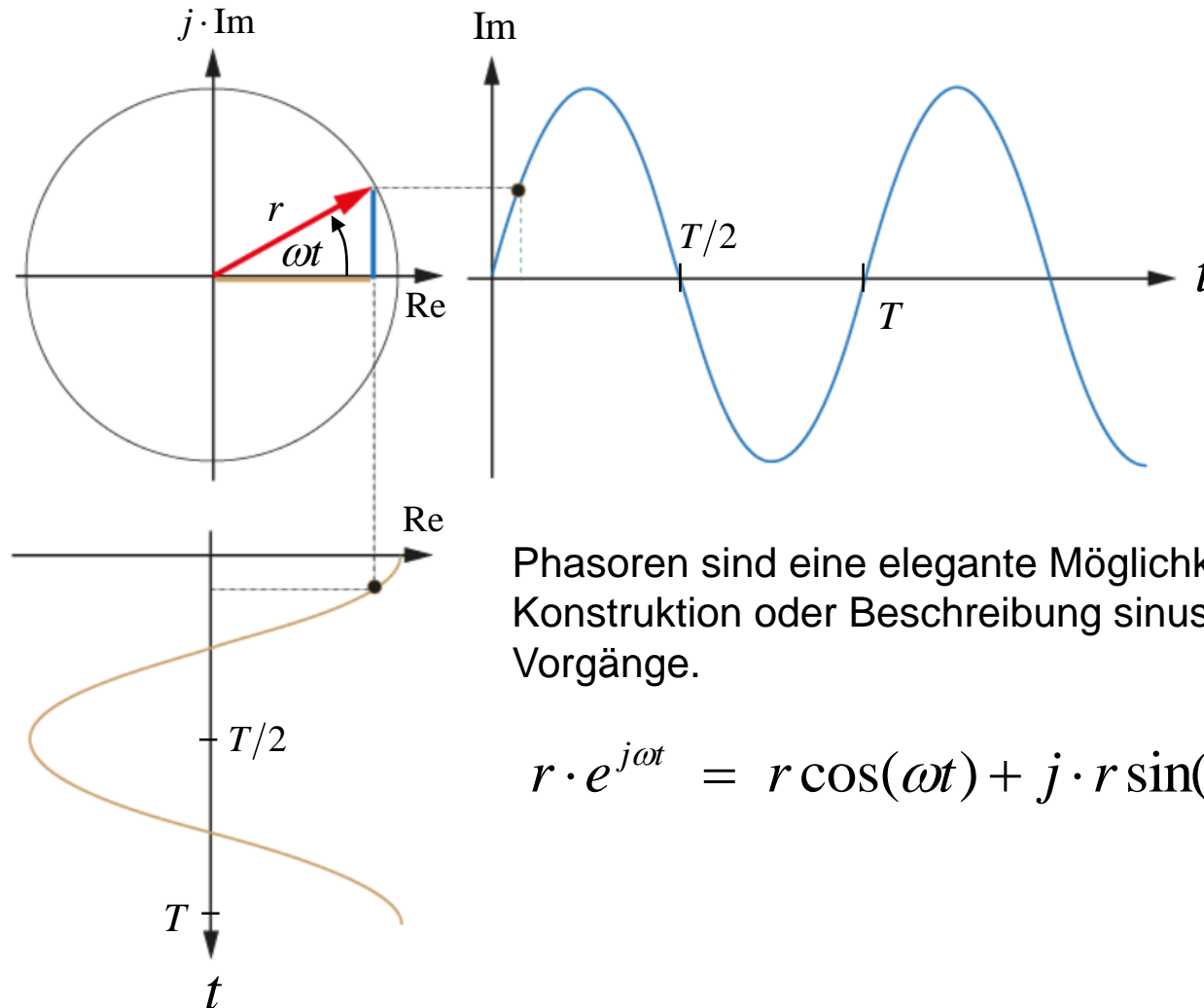
Wenn φ sich zeitabhängig ändert $\varphi = \varphi(t)$, dann
beschreibt der Zeiger eine Kreisbahn mit Radius r .

$r \cdot e^{j2\pi \frac{t}{T}}$ ist somit ein rotierender Zeiger, der in der Zeitspanne T eine
Umdrehung in der komplexen Ebene vollzieht (Radius r).

Andere Schreibweisen sind $r \cdot e^{j2\pi \frac{t}{T}} = r \cdot e^{j2\pi f t} = r \cdot e^{j\omega t}$

mit der Frequenz $f = \frac{1}{T}$ und der Winkelgeschwindigkeit $\omega = 2\pi f$

5 Phasoren und sinusförmigen Wechselgrößen



Phasoren sind eine elegante Möglichkeit zur Konstruktion oder Beschreibung sinusförmiger Vorgänge.

$$r \cdot e^{j\omega t} = r \cos(\omega t) + j \cdot r \sin(\omega t)$$

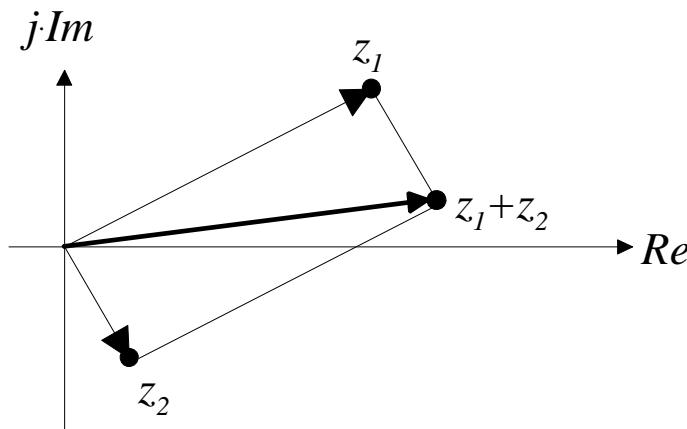


6 Rechnen mit komplexen Zahlen

6.1 Addition/Subtraktion komplexer Zahlen

Für die Addition zweier komplexer Zahlen $\underline{z}_1 = x_1 + jy_1$ und $\underline{z}_2 = x_2 + jy_2$ gilt:

$$\underline{z}_1 + \underline{z}_2 = (x_1 + jy_1) + (x_2 + jy_2) = (x_1 + x_2) + j(y_1 + y_2)$$



Die Addition zweier komplexer Zahlen ist anschaulich eine Zeigeraddition.

Beispiel: $\underline{z}_1 = 3 + j4$ und $\underline{z}_2 = 5 + j6$

$$\underline{z}_1 + \underline{z}_2 = (3 + j4) + (5 + j6) = 8 + j10$$



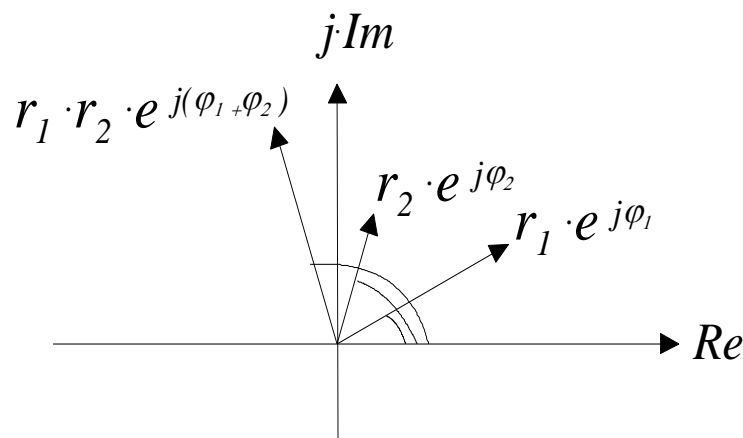
6.2 Multiplikation komplexer Zahlen

Mit Hilfe der Euler-Formel wird auch die Multiplikation komplexer Zahlen sehr einfach geometrisch interpretierbar.

$$\underline{z}_1 \cdot \underline{z}_2 = (x_1 + jy_1) \cdot (x_2 + jy_2) = r_1 e^{j\varphi_1} \cdot r_2 e^{j\varphi_2} = r_1 \cdot r_2 \cdot e^{j(\varphi_1 + \varphi_2)}$$

Die Multiplikation zweier komplexer Zahlen ist anschaulich eine *Drehsteckung*.

- Die Längen der Zeiger werden multipliziert.
- Die Winkel werden addiert





6.3 Division komplexer Zahlen

Mit Hilfe der Euler-Formel wird auch die Division komplexer Zahlen sehr einfach geometrisch interpretierbar.

$$\frac{\underline{z}_1}{\underline{z}_2} = \frac{(x_1 + jy_1)}{(x_2 + jy_2)} = \frac{r_1 e^{j\varphi_1}}{r_2 e^{j\varphi_2}} = \frac{r_1}{r_2} \cdot e^{j(\varphi_1 - \varphi_2)}$$

Die Division zweier komplexer Zahlen ist anschaulich eine *Drehstauchung*.

- Die Längen der Zeiger werden dividiert.
- Die Winkel werden subtrahiert.

7 Komplexe Beschreibung reeller Sinus-/Kosinusfunktionen

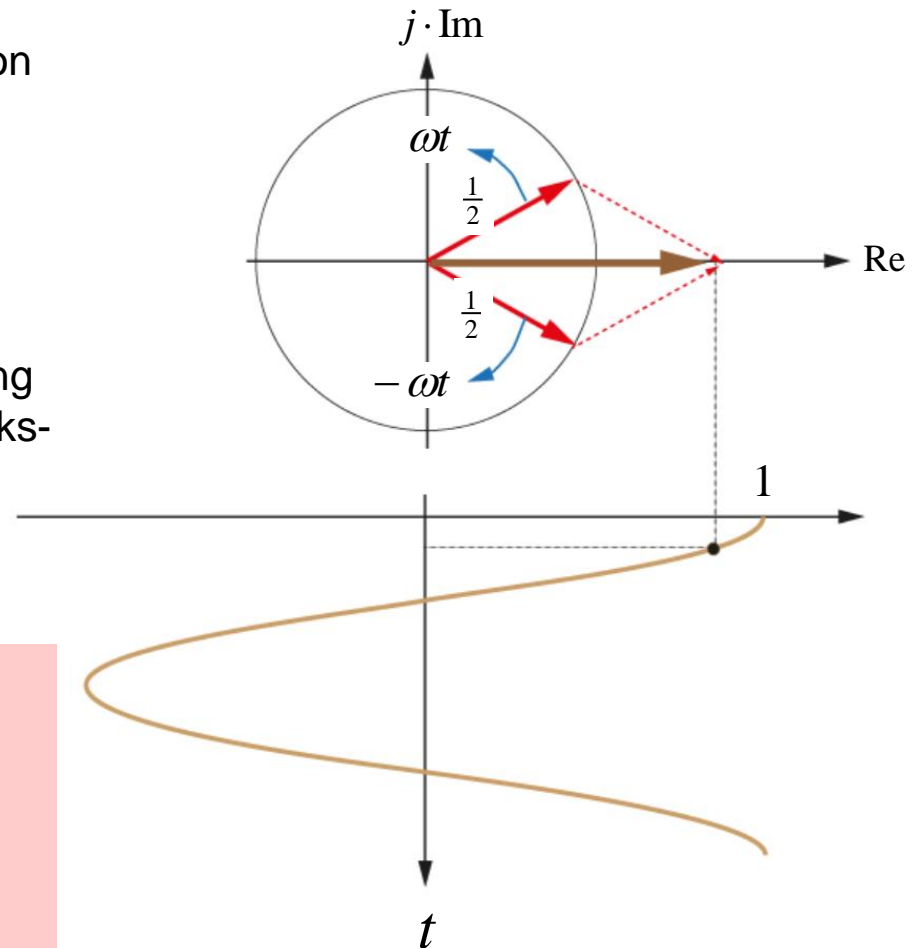
Nach Euler lässt sich eine Kosinusfunktion wie folgt auch komplex beschreiben:

$$\cos(\omega t) = \frac{1}{2} \left[e^{j\omega t} + e^{-j\omega t} \right]$$

Dies folgt unmittelbar aus der Anschauung und ergibt sich aus der Addition eines links- und rechtsdrehenden Zeigers).

Beachte: In der Welt der komplexen Drehzeiger gibt es auch **negative Frequenzen** ($-\omega$).

neg. Frequenz = Zeiger dreht im UZS
pos. Frequenz = Zeiger dreht im GUZS

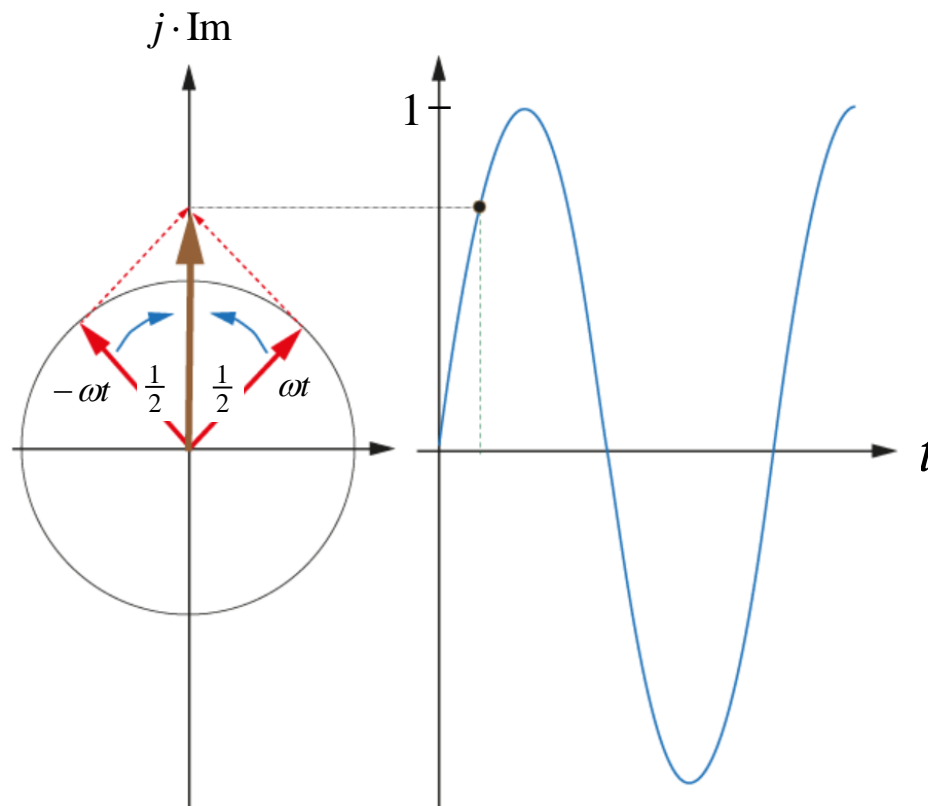




Nach Euler lässt sich eine Sinusfunktion wie folgt auch komplex beschreiben:

$$\sin(\omega t) = \frac{1}{2j} \left[e^{j\omega t} - e^{-j\omega t} \right]$$

Dies folgt unmittelbar aus der Anschauung und ergibt sich ebenfalls aus der Addition eines links- und rechtsdrehenden (negativen) Zeigers.

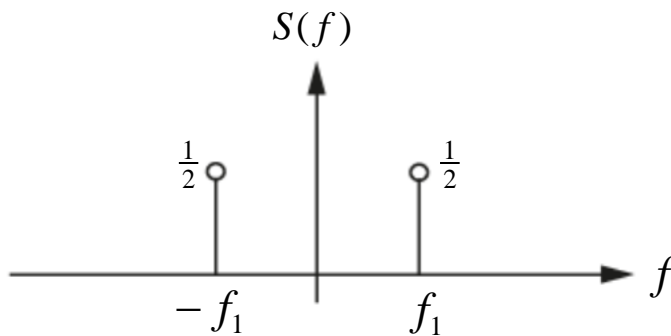




8 Darstellung von Sinus- und Kosinussignalen im Frequenzbereich

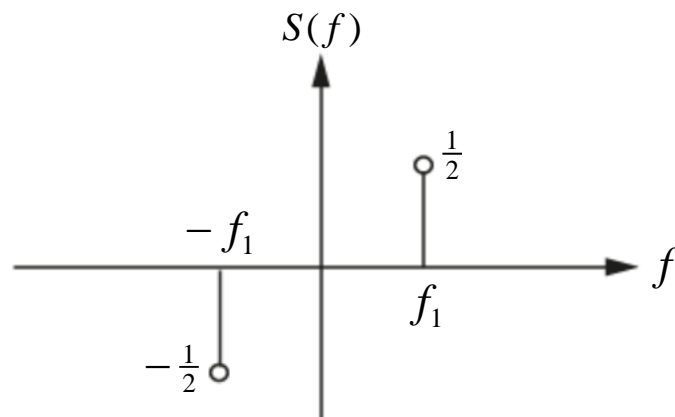
Um die Frequenzzusammensetzung eines Signals darzustellen verwendet man oft die „Darstellung im Frequenzbereich“.

$S(f)$ ist die Amplitude in Abhängigkeit von der Frequenz.



$$\cos(\omega_1 t) = \frac{1}{2} \left[e^{j\omega_1 t} + e^{-j\omega_1 t} \right] \quad \omega_1 = 2\pi f_1$$

Je höher die Frequenz des Signals ist, desto mehr rücken die Frequenzstriche nach außen.



$$\sin(\omega_1 t) = \frac{1}{2j} \left[e^{j\omega_1 t} - e^{-j\omega_1 t} \right]$$



Übung: Nichtlineares System

Gegeben ist ein nichtlineares System, mit folgendem Verhalten: $g(t) = s^2(t)$

Welches Signal $g(t)$ wird ausgegeben, wenn für das Eingangssignal gilt:

- a) $s(t) = \cos(\omega t)$
- b) $s(t) = 1 + \sin(\omega t)$



9 Zusammenfassung : *Signalbeschreibung mit Phasoren*

Sinusförmige Signale sind die Basis der Signalverarbeitung, da sich alle Signale als Überlagerung von sinusförmigen Signalen beschreiben lassen (folgt später).

Sinusförmige Signale werden bevorzugt durch Phasoren (komplexe Drehzeiger) beschrieben.

Grund: Addition, Subtraktion, Multiplikation und Division von Signalen sind wesentlich einfacher zu berechnen (anschauliche Zeigeroperationen) wenn man anstelle von Sinus- und Kosinusfunktionen die korrespondierenden Phasoren verwendet.



A

2. Mathematische Grundlagen und Historie

- Komplexe Zahlen – eine kurze Wdh.
- **Fourierreihe**
- Diskrete Form der Fourierreihe

1 Synthese analytischer periodischer Funktionen

Joseph de Fourier entdeckte 1807, das sich (alle) periodischen Funktionen $s(t)$ als trigonometrische Funktionenreihe darstellen lassen⁽¹⁾:

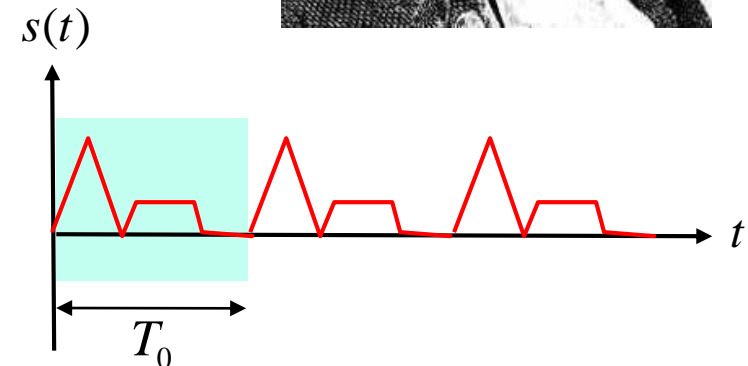
$$s(t) = \frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k \cos k\omega_0 t + b_k \sin k\omega_0 t) \quad (1)$$

mit $\omega_0 = \frac{2\pi}{T_0} = 2\pi \cdot f_0$

T_0 Periodendauer von $s(t)$

$f_0 = \frac{1}{T_0}$ Grundfrequenz

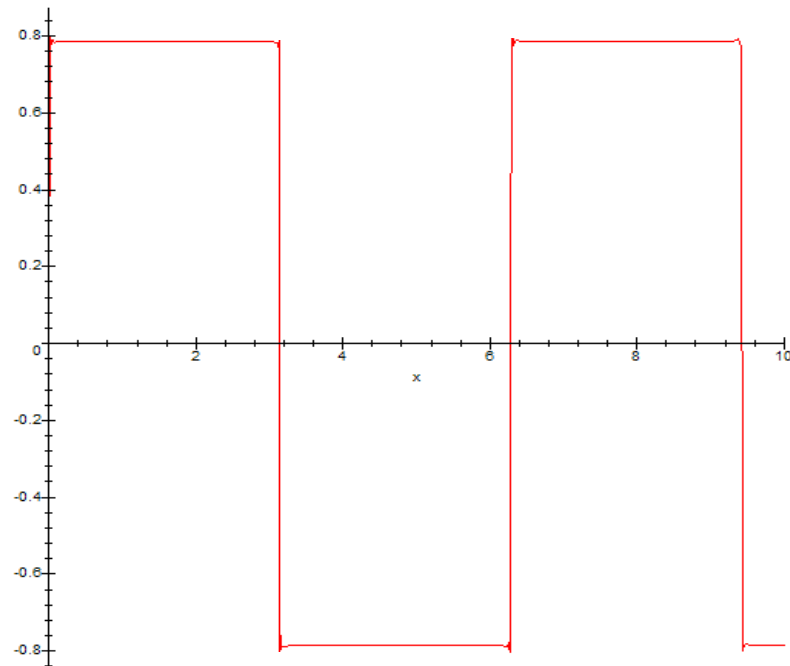
a_k, b_k Fourierkoeffizienten (Wichtungsfaktoren der Vielfachen der Grundfrequenz f_0)



(1) Voraussetzung (Dirichletsche Bedingung): $f(x)$ ist in allen Teilintervallen im Bereich $0..2\pi$ stetig und monoton.



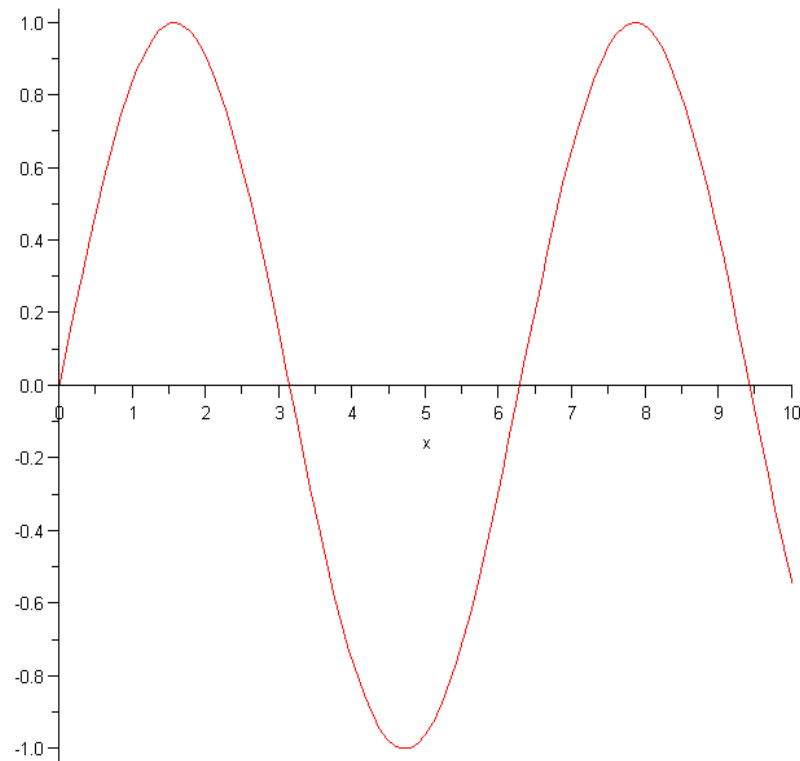
Beispiel: Approximation der folgenden Rechteckfunktion





Grundschwungung

$$s(t) = \sin(t)$$



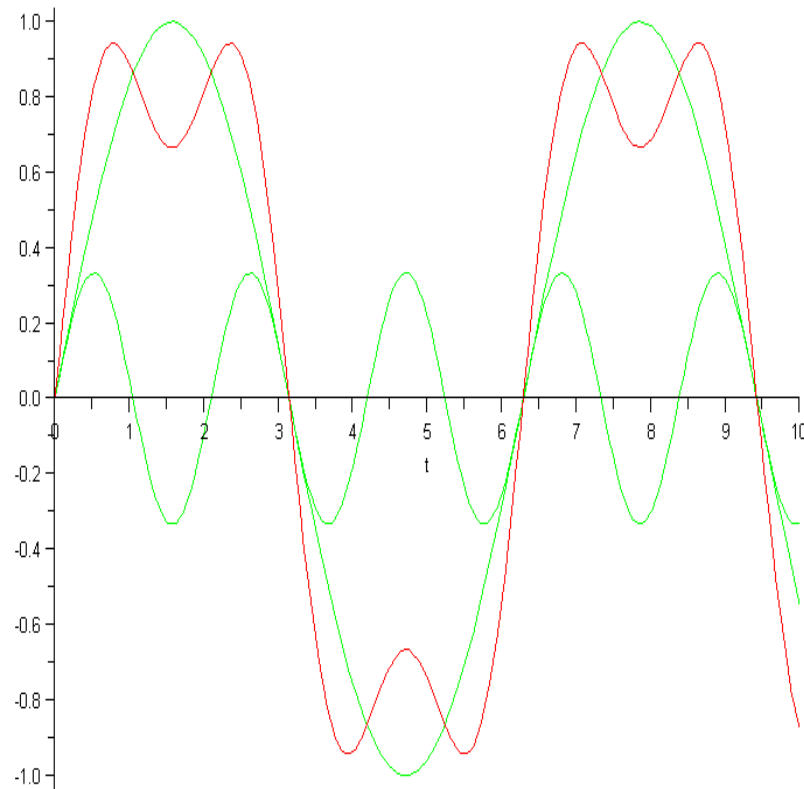
$$a_0 = 0$$

$$a_1 = 0$$

$$b_1 = 1$$



$$s(t) = \sin(t) + \frac{\sin(3t)}{3}$$



$$a_0 = 0$$

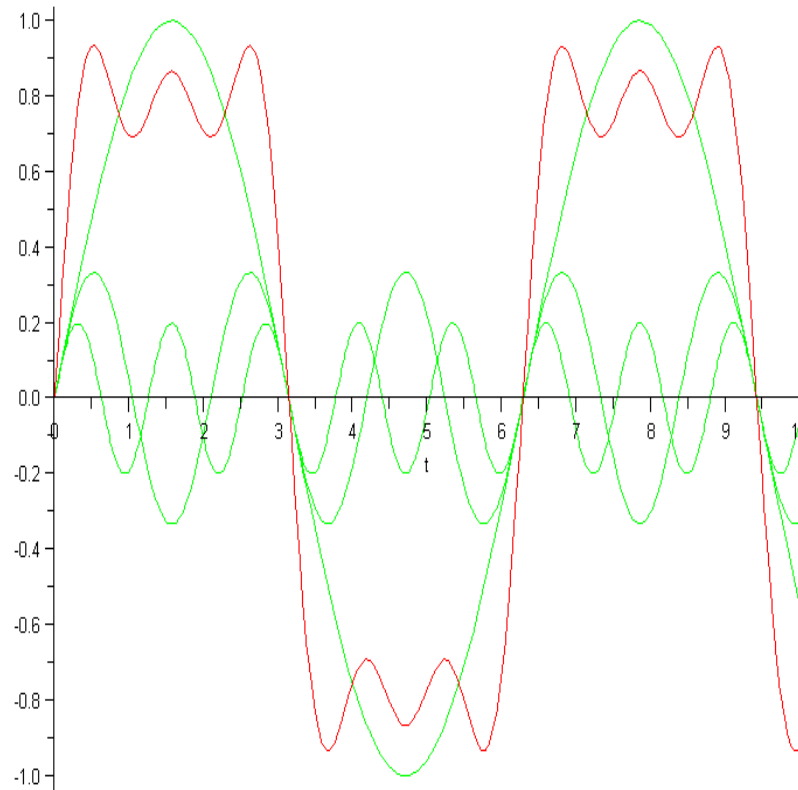
$$a_1 = 0 \quad b_1 = 1$$

$$a_2 = 0 \quad b_2 = 0$$

$$a_3 = 0 \quad b_3 = \frac{1}{3}$$



$$s(t) = \sin(t) + \frac{\sin(3t)}{3} + \frac{\sin(5t)}{5}$$



$$a_0 = 0$$

$$a_1 = 0 \quad b_1 = 1$$

$$a_2 = 0 \quad b_2 = 0$$

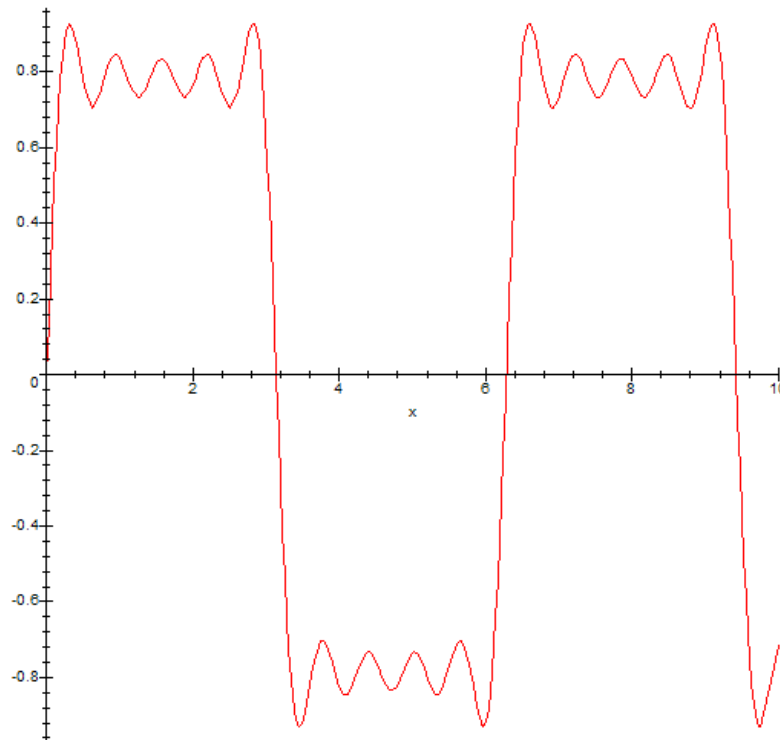
$$a_3 = 0 \quad b_3 = \frac{1}{3}$$

$$a_4 = 0 \quad b_4 = 0$$

$$a_5 = 0 \quad b_5 = \frac{1}{5}$$



$$s(t) = \sin(t) + \frac{\sin(3t)}{3} + \frac{\sin(5t)}{5} + \frac{\sin(7t)}{7} + \frac{\sin(9t)}{9}$$



$$a_0 = 0$$

$$a_1 = 0 \quad b_1 = 1$$

$$a_2 = 0 \quad b_2 = 0$$

$$a_3 = 0 \quad b_3 = \frac{1}{3}$$

$$a_4 = 0 \quad b_4 = 0$$

$$a_5 = 0 \quad b_5 = \frac{1}{5}$$

$$a_6 = 0 \quad b_6 = 0$$

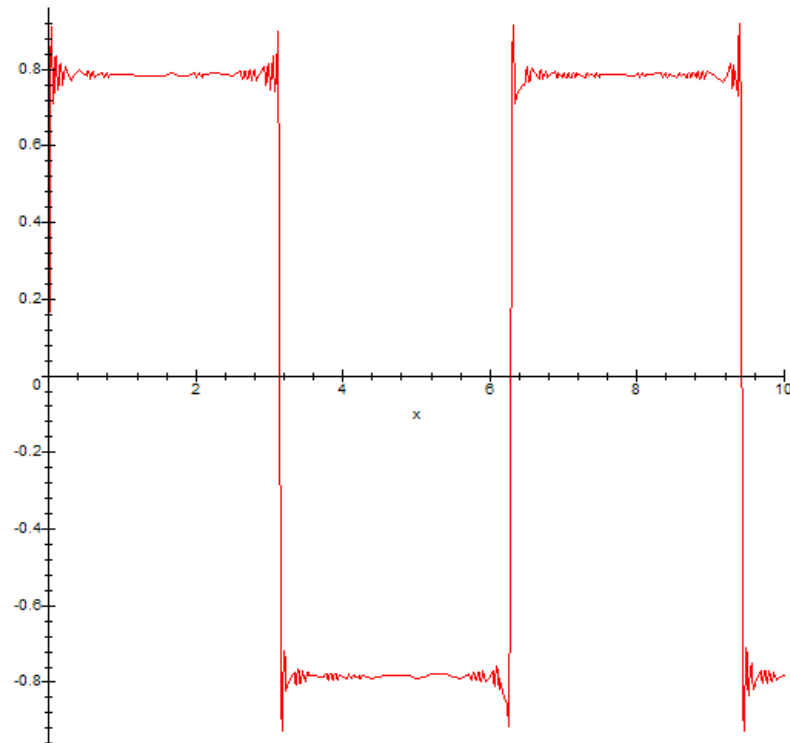
$$a_7 = 0 \quad b_7 = \frac{1}{7}$$

$$a_8 = 0 \quad b_8 = 0$$

$$a_9 = 0 \quad b_9 = \frac{1}{9}$$

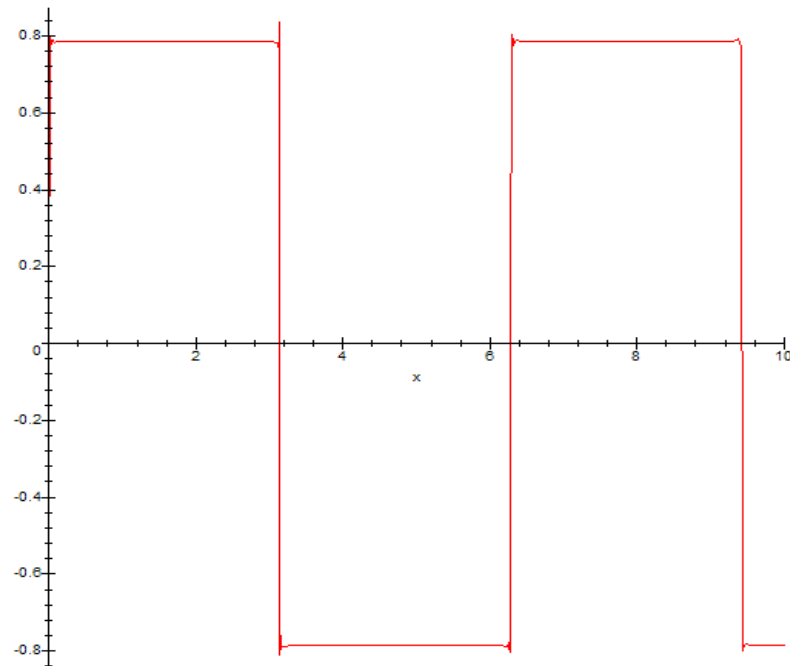


$$s(t) = \sin(t) + \frac{\sin(3t)}{3} + \frac{\sin(5t)}{5} + \frac{\sin(7t)}{7} + \frac{\sin(9t)}{9} + \dots + \frac{\sin(99t)}{99}$$



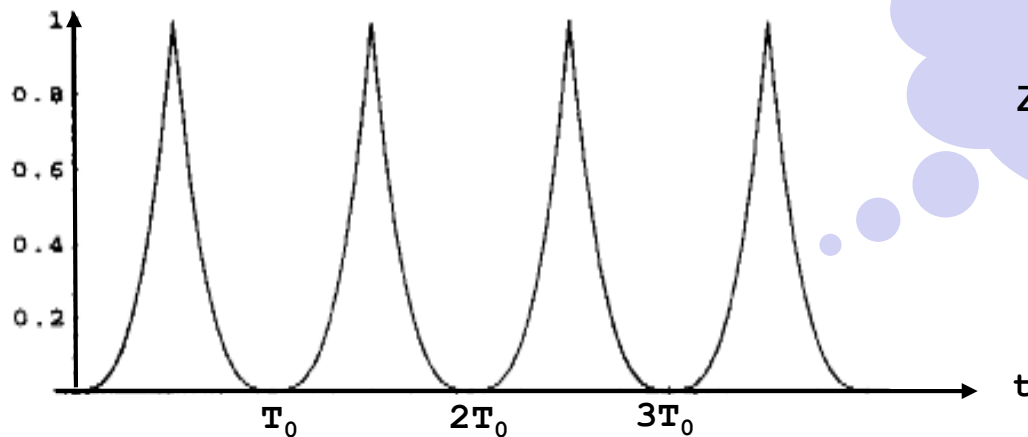


$$s(t) = \sin(t) + \frac{\sin(3t)}{3} + \frac{\sin(5t)}{5} + \frac{\sin(7t)}{7} + \frac{\sin(9t)}{9} + \dots + \frac{\sin(999t)}{999}$$



2 Analyse analytischer periodischer Funktionen

Gegeben sei eine periodische Funktion.



Aus welchen
sin-/cos-Funktionen
ist diese Funktion
zusammengesetzt ?

oder präziser gefragt:

Wie müssen die Wichtungsfaktoren der Sinusfunktionen (b_k) und der Cosinusfunktionen (a_k) gewählt werden, damit genau diese periodische Funktion entsteht ?

$$s(t) = \frac{a_0}{2} + a_1 \cos(\omega_0 t) + a_2 \cos(2\omega_0 t) + a_3 \cos(3\omega_0 t) + \dots \\ + b_1 \sin(\omega_0 t) + b_2 \sin(2\omega_0 t) + b_3 \sin(3\omega_0 t) + \dots$$



Die Fourierkoeffizienten einer periodischen Funktion $s(t)$ können wie folgt berechnet werden (o. Bew.):

$$a_0 = \frac{2}{T_0} \int_0^{T_0} s(t) dt$$

$$a_k = \frac{2}{T_0} \int_0^{T_0} s(t) \cos(k\omega_0 t) dt$$

$$b_k = \frac{2}{T_0} \int_0^{T_0} s(t) \sin(k\omega_0 t) dt$$



3 Komplexe Darstellung (eleganter)

Meist fasst man a_k und b_k wie folgt zu komplexen Koeffizienten \underline{c}_k zusammen:

$$\underline{c}_k = \frac{1}{2} (a_k - j b_k) \quad (5)$$

Man erhält somit eine gemeinsame Darstellung für die sin- und cos-Komponenten

$$\begin{aligned} \underline{c}_k &= \frac{1}{T_0} \int_0^{T_0} s(t) \cos(k\omega_0 t) dt - j \frac{1}{T} \int_0^{T_0} s(t) \sin(k\omega_0 t) dt \\ &= \frac{1}{T_0} \int_0^{T_0} s(t) \cdot [\cos(k\omega_0 t) - j \sin(k\omega_0 t)] dt \\ &= \frac{1}{T_0} \int_0^{T_0} s(t) \cdot e^{-jk\omega_0 t} dt \end{aligned}$$



4 Zusammenfassung : Fourierreihe

- analytisches Verfahren für kontinuierliche periodische Funktionen (Signale)
- für die Bestimmung der Koeffizienten sind Integrale zu lösen
- unpraktisch für die Anwendung auf digitale Signale (Abtastwerte)



A

2. Mathematische Grundlagen und Historie

- Komplexe Zahlen – eine kurze Wdh.
- Fourierreihe
- **Diskrete Form der Fourierreihe**

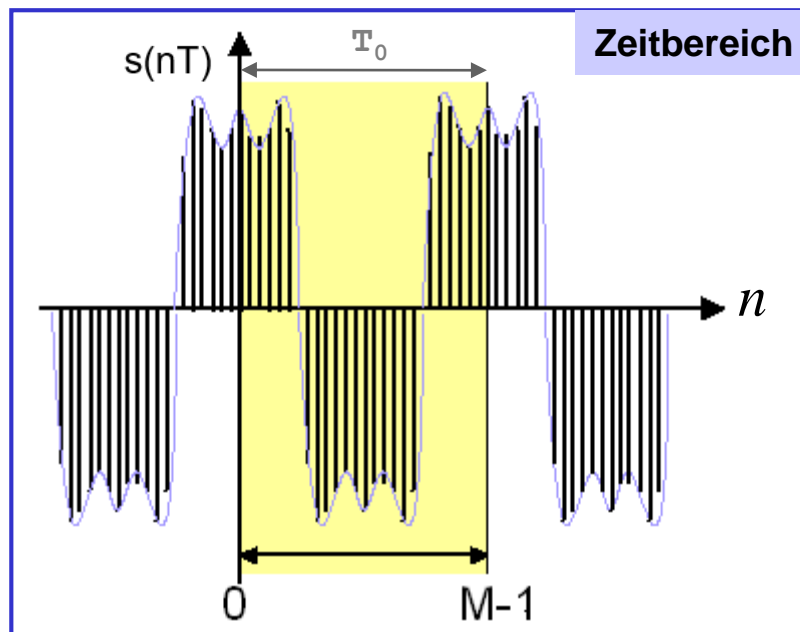


1 Analyse abgetasteter periodischer Funktionen

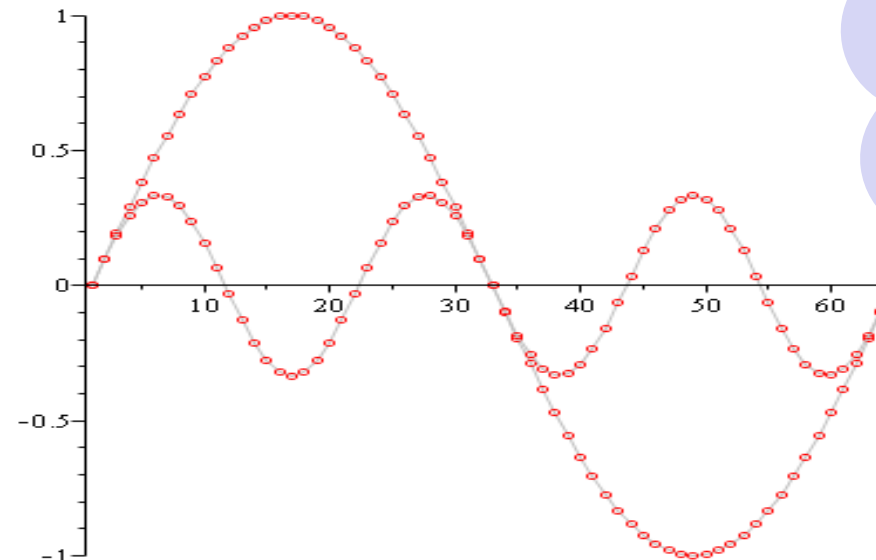
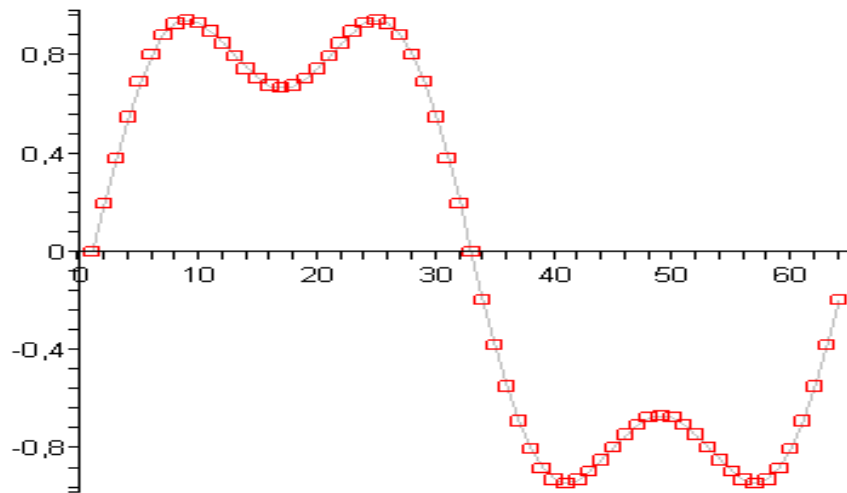
1.1 Fragestellung

Gegeben: eine Periode eines abgetasteten periodischen Signals $s(nT)$

- bestehend aus M Abtastwerten im Abstand T und
- mit der Grundfrequenz f_0 .



Aus welchen (abgetasteten) sin-/cos-Schwingungen der Frequenz $k \cdot f_0$ ($k=1,2,3,\dots$) ist das Signal zusammengesetzt ?



.... oder anders gefragt :

Mit welchen Wichtungen $a(k)$ und $b(k)$ sind (abgetastete) cos- und sin-Schwingungen der Frequenz kf_0 ($k=1,2,3,\dots$) in einem periodischen Signal $s(t)$ enthalten ?

z.B. hier:

$$s(t) = \sin(2\pi \cdot 1 \cdot \frac{t}{M}) + \frac{1}{3} \sin(2\pi \cdot 3 \cdot \frac{t}{M})$$

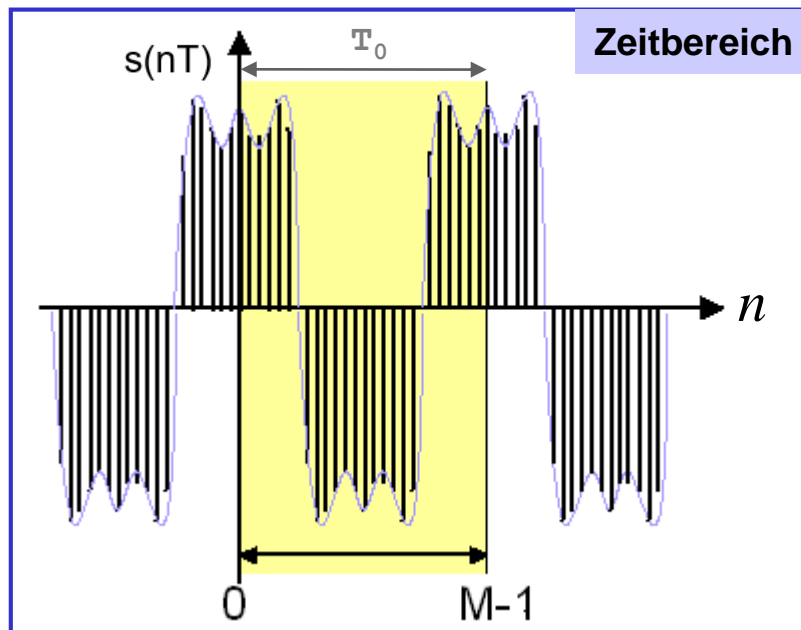
$$t = 0, 1, 2, 3, 4, \dots, M-1$$

1.2 Periodendauer, Abtastintervall und Zahl der Abtastwerte

Zwischen der Periodendauer T_0 und der Grundfrequenz f_0 gilt: $f_0 = \frac{1}{T_0}$

T ist der Abstand zweier aufeinanderfolgender Abtastwerte.

Daraus folgt für die Anzahl der Abtastwerte: $M = \frac{T_0}{T}$



1.3 Diskrete Fouriertransformation

= Analyse eines abgetasteten, periodischen Signals $s(nT)$

Die Wichtungsfaktoren $a(k)$ der cos-Funktionen werden bestimmt mit:

$$a_k = \frac{2}{T_0} \cdot \sum_{n=0}^{M-1} s(nT) \cdot \cos\left(2\pi \cdot \frac{k}{M} \cdot n\right)$$

$$k = 0, 1, 2, 3, \dots, M-1$$

Die Wichtungsfaktoren $b(k)$ der sin-Funktionen werden bestimmt mit:

$$b_k = \frac{2}{T_0} \cdot \sum_{n=0}^{M-1} s(nT) \cdot \sin\left(2\pi \cdot \frac{k}{M} \cdot n\right)$$

M : Anzahl der Abtastwerte

$s(nT)$: abgetastetes Signal (bei nT)

k : Vielfache der Grundfrequenz

n : Abtastwertindex

→ die DFT beschreibt die Zerlegung einer abgetasteten, periodischen Funktion in (abgetastete) sin-/cos-Funktionen (=Analyse)

1.4 Normierte Form der diskreten Fouriertransformation

Zwischen der Periodendauer T_0 , dem Abtastintervall T und der Anzahl der Abtastwerte M gilt:

$$M = \frac{T_0}{T}$$

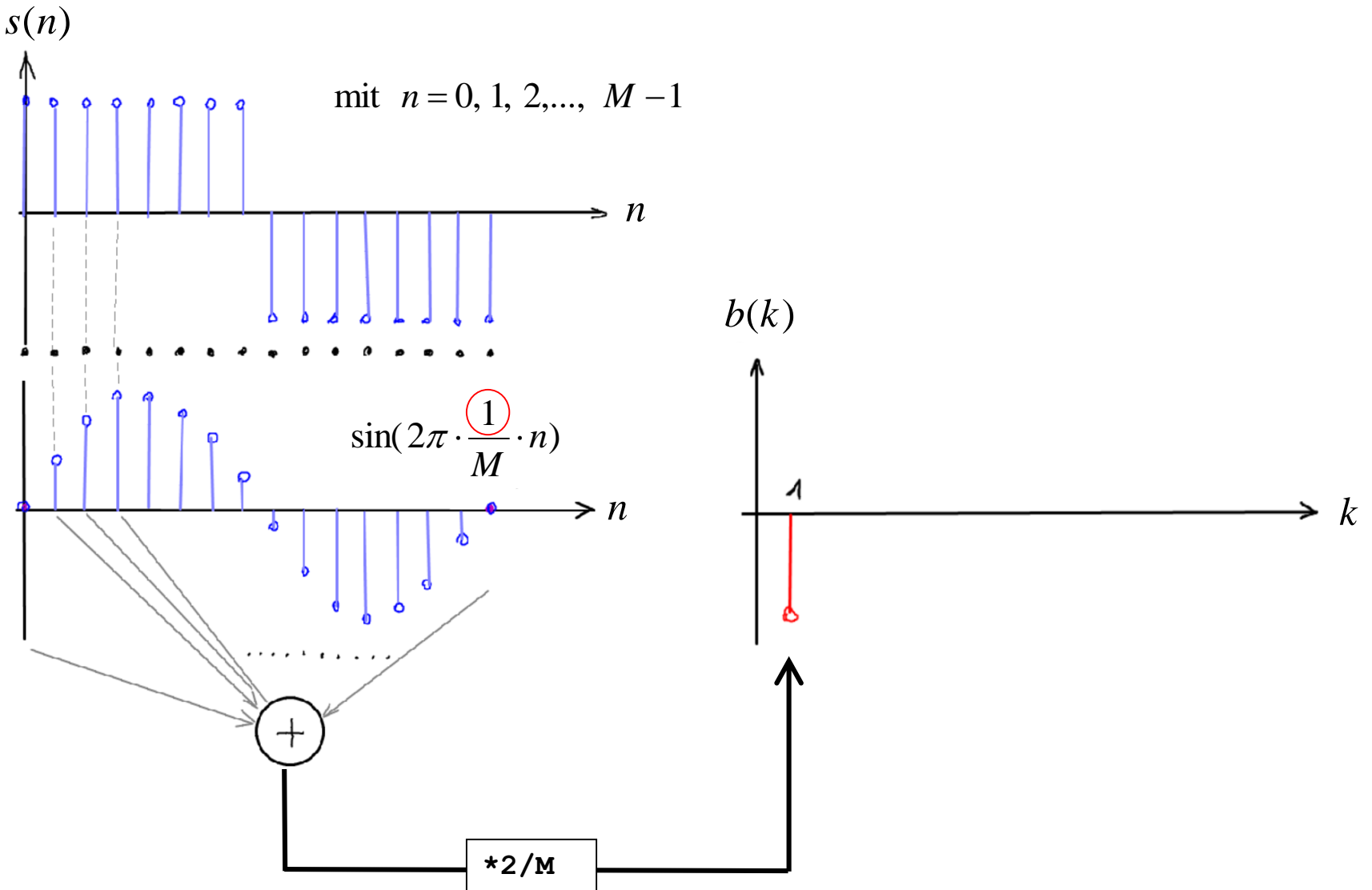
Normiert man die Abtastzeit auf $T=1$ dann gilt $T_0 = M$ und die Transformation vereinfacht sich zu (zeitfrei):

$$a_k = \frac{2}{T_0} \cdot \sum_{n=0}^{M-1} s(nT) \cdot \cos\left(2\pi \cdot \frac{k}{M} \cdot n\right) \quad \Rightarrow \quad a_k = \frac{2}{M} \cdot \sum_{n=0}^{M-1} s(n) \cdot \cos\left(2\pi \cdot \frac{k}{M} \cdot n\right)$$

$$b_k = \frac{2}{T_0} \cdot \sum_{n=0}^{M-1} s(nT) \cdot \sin\left(2\pi \cdot \frac{k}{M} \cdot n\right) \quad \Rightarrow \quad b_k = \frac{2}{M} \cdot \sum_{n=0}^{M-1} s(n) \cdot \sin\left(2\pi \cdot \frac{k}{M} \cdot n\right)$$

Diese Form der DFT transformiert eine Sequenz von M Abtastwerten in zwei Sequenzen von M Wichtungsfaktoren.

Beispiel: DFT – Berechnung von b_1





ÜBUNG: Diskrete Fourier-Transformation

Geben Sie einen Algorithmus für die DFT an.



2 Synthese abgetasteter periodischer Funktionen

2.1 Inverse Diskrete Fouriertransformation

Die IDFT beschreibt die Synthese einer periodischen, abgetasteten Funktion aus gewichteten (mit $a(k)$ und $b(k)$), abgetasteten cos- und sin-Funktionen.

Aus der Fourierreihe wird durch Ersetzen der Integrale durch Summen

$$s(nT) = \frac{a_0}{2} + \sum_{k=0}^{M-1} \left[a_k \cdot \cos\left(2\pi \cdot \frac{k}{T_0} \cdot nT\right) + b_k \cdot \sin\left(2\pi \cdot \frac{k}{T_0} \cdot nT\right) \right]$$

mit $M = \frac{T_0}{T}$ wird daraus kürzer

$$s(nT) = \frac{a_0}{2} + \sum_{k=0}^{M-1} \left[a_k \cdot \cos\left(2\pi \cdot \frac{n}{M} \cdot k\right) + b_k \cdot \sin\left(2\pi \cdot \frac{n}{M} \cdot k\right) \right]$$

mit $n = 0, 1, 2, \dots, M-1$



2.2 Normierte Formen der Inversen Diskreten Fouriertransformation

oder mit der Normierung $T=1$ $\rightarrow M = \frac{T_0}{T} = T_0$

$$s(n) = \frac{a_0}{2} + \sum_{k=0}^{M-1} \left[a_k \cdot \cos\left(2\pi \cdot \frac{n}{M} \cdot k\right) + b_k \cdot \sin\left(2\pi \cdot \frac{n}{M} \cdot k\right) \right]$$

oder mit der Normierung $T_0=1$ (und damit $f_0=1$) $\rightarrow M = \frac{T_0}{T} = \frac{1}{T} = f_s$

$$s(nT) = \frac{a_0}{2} + \sum_{k=0}^{M-1} \left[a_k \cdot \cos(2\pi k \cdot nT) + b_k \cdot \sin(2\pi k \cdot nT) \right]$$



ÜBUNG: Inverse diskrete Fourier-Transformation

Geben Sie einen Algorithmus für die IDFT an.



3 Komplexe Darstellung

Meist fasst man $a(k)$ und $b(k)$ wie folgt zu komplexen Koeffizienten $\underline{c}(k)$ zusammen:

$$\underline{c}_k = \frac{1}{2} [a_k - j \cdot b_k]$$

Man erhält somit eine gemeinsame Darstellung für die sin- und cos-Komponenten des Signals

$$\begin{aligned}\underline{c}_k &= \frac{1}{T_0} \cdot \sum_{n=0}^{M-1} s(nT) \cos(2\pi \frac{k}{M} \cdot n) - j \frac{1}{T_0} \sum_{n=0}^{M-1} s(nT) \sin(2\pi \frac{k}{M} \cdot n) \\ &= \frac{1}{T_0} \cdot \sum_{n=0}^{M-1} s(nT) \left[\cos(2\pi \frac{k}{M} \cdot n) - j \cdot \sin(2\pi \frac{k}{M} \cdot n) \right] \\ &= \frac{1}{T_0} \cdot \sum_{n=0}^{M-1} s(nT) \cdot e^{-j2\pi \frac{k}{M} \cdot n}\end{aligned}$$



oder normiert mit $T=1$

$$\text{d.h. } M = \frac{T_0}{T} = T_0$$

$$\underline{c}_k = \frac{1}{M} \cdot \sum_{n=0}^{M-1} s(n) \cdot e^{-j2\pi \frac{k}{M} \cdot n}$$

oder normiert mit $T_0=1$

$$\text{d.h. } M = \frac{T_0}{T} = \frac{1}{T} = f_s$$

$$\underline{c}_k = \sum_{n=0}^{M-1} s(nT) \cdot e^{-j2\pi k \cdot nT}$$



4 Zusammenfassung : Diskrete Fouriertransformation

- numerisches Verfahren für diskrete periodische Funktionen (Signale)
- für die Bestimmung der Koeffizienten sind Summen zu berechnen
- ideal für die Anwendung auf digitale Signale (Abtastwerte)

Statt des gezeigten DFT-Algorithmus wird sinnvollerweise eine stark optimierte Variante verwendet, die sog. **Fast Fourier Transformation** (FFT). → folgt später

5 Historische Kuriosität

Als „*Erfinder*“ der **FFT** werden meist Cooley und Tukey genannt (1965).

Tatsächlich hat Carl Friedrich Gauss 1805 die **DFT** im Zusammenhang mit der Berechnung von Asteroiden-Orbits entwickelt. Für die numerische Umsetzung entwickelte er einen **FFT**-Algorithmus. Es wurde jedoch von ihm nie veröffentlicht.

Das war einige Jahre vor Fourier (analytische Version) und 160 Jahre vor Cooley und Tukey.

