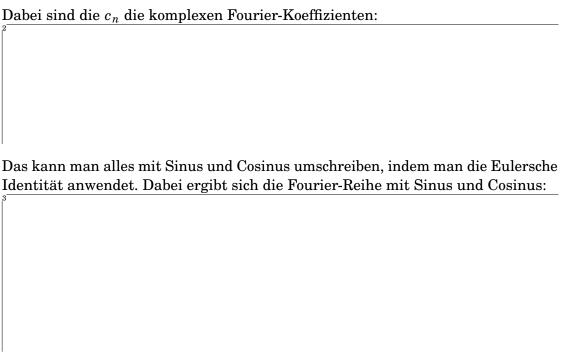
17

Fourier-Reihe mit Sinus und Cosinus. Fast Fourier Transform

Jörn Loviscach
Versionsstand: 21. März 2014, 21:10
Die nummerierten Felder sind absichtlich leer, zum Ausfüllen beim Ansehen der Videos: http://www.j3L7h.de/videos.html
This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Germany License. To view a copy of the license, visit http://creativecommons.org/licenses/by-nc-sa/3.0/de/ or send a letter to Creative Commons, 171 Second Street, Suite 300, Strancisco, California, 94105, USA.
Bitte hier notieren, was beim Bearbeiten unklar geblieben ist

1 Fourier-Reihe mit Sinus und Cosinus

Gegeben sei eine reell- oder komplexwertige Funktion f mit Periode T . Dann kai				
	man die in eine komplexe Fourier-Reihe entwickeln:			
	1			



Nicht wundern: Dass a_0 mit dem Faktor 1/2 steht, macht nachher einige Formeln einfacher. Und Vorsicht mit der Reihenfolge: Die a_n stehen mit dem Cosinus.

Zwei Unterschiede zur komplexen Fourier-Reihe:

- Es tauchen keine negativen Frequenzen mehr auf.
- Die Amplitude und die Phase der n-ten Oberwelle ist nun in a_n und b_n versteckt statt im Betrag und Winkel von c_n .

Ist f eine gerade (und weiterhin periodische!) Funktion, d. h. f(-t) = f(t) für alle t, dann kann offensichtlich kein Sinus vorkommen, also sind alle b_n gleich null. Umgekehrt kann in einer ungeraden Funktion f, d. h. f(-t) = -f(t) für alle t, kein Cosinus vorkommen, also sind alle a_n gleich null.

2 Fourier-Koeffizienten für Sinus und Cosinus

Man könnte aus c_n und c_{-n} die Fourier-Koeffizienten a_n und b_n bestimmen. Aber es gibt auch einen direkten Weg. Dazu überlegt man sich Folgendes:

$$\int_{0}^{T} (\cos(2\pi nt/T))^{2} dt = \int_{0}^{4} \text{ für alle } n = 1, 2, 3, ...,$$

denn:	
6	
$\int_0^T (\sin(2\pi nt/T))^2 dt = $	£:11 1 0 0
$\int_0^{\infty} \left(\sin(2\pi nt/T) \right) at = \int_0^{\infty} \left \sin(2\pi nt/T) \right dt$	für alle $n = 1, 2, 3,,$
denn:	
7	
<u></u>	
f^T	
$\int_{0}^{T} \sin(2\pi nt/T) \sin(2\pi mt/T) dt = \int_{0}^{T} \sin(2\pi nt/T) dt$	für alle $n, m = 1, 2, 3, \dots$ mit $n \neq m$
denn mit zweifacher partieller Integration	ergibt sich:
Und entsprechend:	
•	
$_{m C}T$	
$\int_0^{\infty} \sin(2\pi nt/T)\cos(2\pi mt/T)dt = \int_0^{\infty} \sin(2\pi nt/T)dt = \int_0^{\infty} \sin($	für alle $n, m = 1, 2, 3, \dots$ mit $n \neq m$
- 0	Tota matical acceptance
und das sogar für $n = m$ (einfache partielle	Integration) sowie
cT	
$\int_0^T \cos(2\pi nt/T)\cos(2\pi mt/T)dt =$	für alle $n, m = 1, 2, 3, \dots$ mit $n \neq m$
J0	
Daran sieht man, dass sich die Sinus- und	Cosinus-Funktionen der Fourier-Reihe
fast so verhalten wie die Funktionen $t \mapsto e^2$	
stehen senkrecht aufeinander und haben al	
Wenn man also $f(t)$ mit $\cos(2\pi mt/T)$ integri	tert ($m = 1, 2, 3,$), wird man erhalten:

Also muss für die Fourier-Koeffizienten a_n gelten:
Durch den Trick mit dem $\frac{1}{2}a_0$ gilt diese Formel auch für den Gleichspannungsan-
teil $n = 0$. Entsprechend gibt für die Fourier-Koeffizienten b_n :
3 Fast Fourier Transform (FFT)
Typischerweise hat man Signale als Folgen von Messwerten (Samples) gegeben statt als kontinuierliche Funktionen (Demo mit Audacity).
Dann lassen sich die Fourier-Koeffizienten mit Summen über die Samples statt mit Integralen berechnen (Diskrete Fourier-Transformation, DFT). Indem man diese Summen geschickt zusammenfasst, kann man die Rechnung beschleunigen (Fast Fourier Transform, FFT). Dies ist die übliche Art, Fourier-Analyse zu betreiben. Dafür stehen auch diverse Programmierbibliotheken bereit, insbesondere FFTW (Link).
DFT/FFT beziehen sich immer nur auf einen endlichen Ausschnitt eines Signals (Demo mit Audacity). Diese Ausschnitte werden weich mit "Fensterfunktionen" [window functions] aus dem Signal gebildet. Harte Schnitte würden unsinnige Anteile mit hohen Frequenzen in der DFT/FFT erzeugen:

 $\mathrm{MATLAB}^{\$}$ beherrscht selbstverständlich FFT und Fensterfunktionen.