

Assignment 3: Fourier Transformation in Matlab

Angewandte Modellierung 25

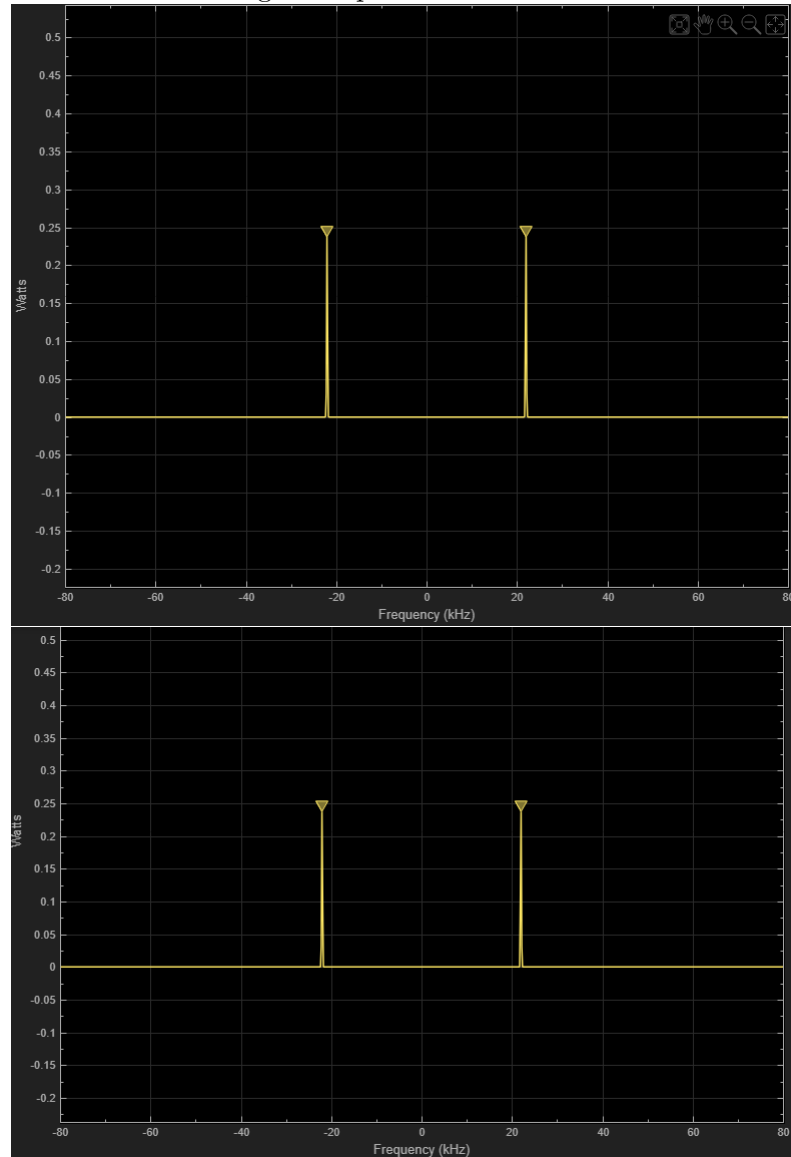
Carl Colmant

May 9, 2025

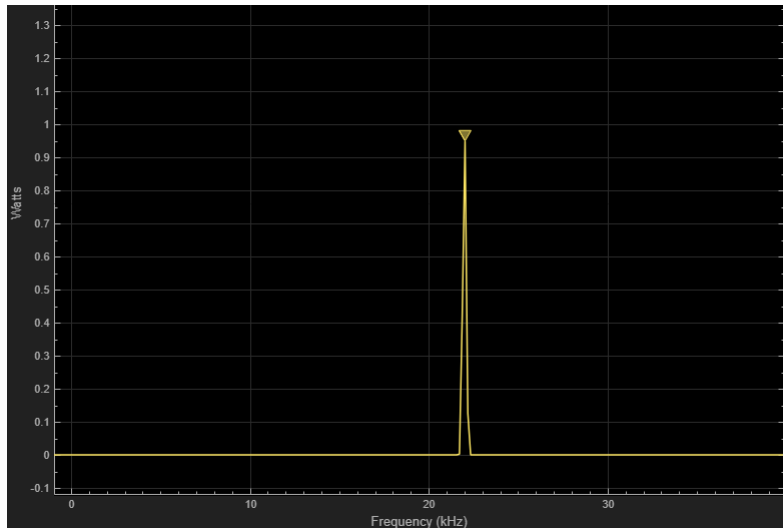
Exercise 1. Modellierung von Sinussignalen

Zu erst habe ich die Blöcke wie in der Aufgabe beschrieben erstellt (meine letzte Ziffer der matrikelnr. ist 2).

Dabei entstehen folgende Spektren:



Und das Spectrum der Summe der Signale:



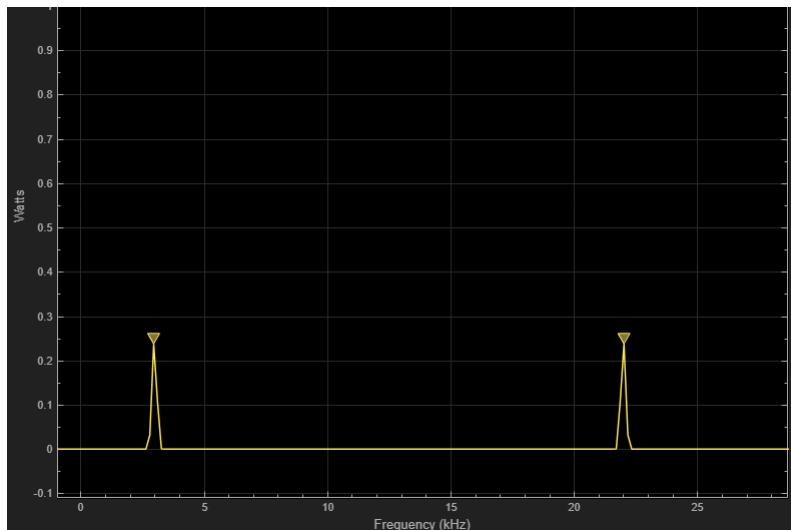
Die beiden Sinus funktionen haben die Frequenz 22000Hz die erste wird um $\frac{\pi}{2}$ verschoben heißt also $\sin(22000 \cdot t + \frac{\pi}{2})$ und die zweite ist $\sin(22000 \cdot t)$. Weil die beiden Signale die gleiche Frequenz haben, bekommen wir im Spectrum Analyzer den selben Peak bei 22000Hz und -22000Hz.

Die zweite Sinus Funktion wird vor dem Addieren noch mit j multipliziert. Das heißt wir addieren $\cos(22000t) + j * \sin(22000t) = e^{j*22000t}$ und das ist die Darstellung der komplexen Exponentialfunktion. Die nur eine Spectrallinie hat bei 22000Hz.

Das kann auch in der Simulation gesehen werden.

Frequenz Veränderung

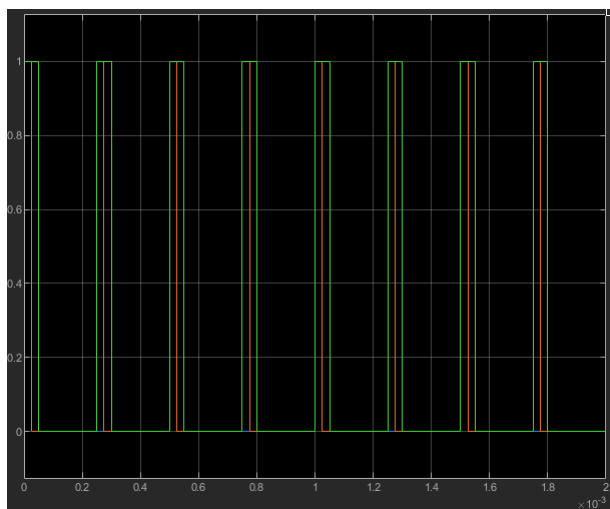
Die Frequenz der Sinus Signale wird in Block Sine Wave 1 auf $2\pi * 3000$ verringert. Nun haben wir keine gleichen Frequenzen der beiden Signale das heißt es entsteht ein überlagertes komplexes Signal, mit 2 Spectral linien.



Phasen Verschiebung

Wenn man die Phase im zweiten Sine wave Block auf 4 setzt ergibt sich kein Unterschied im Spectrum. Das liegt daran das sich die Phase nicht auf die Frequenz auswirkt.

Exercise 2: Analyse von puls Generator Spektren



Wenn man sich den Scope ansieht sieht man das der dritte und vierte Puls Block die doppelte Frequenz hat wie der erste und zweite. Nun ist die Pulse width im ersten block 5% und im zweiten 10%. Der dritte block hat die doppelte Pulse width wie der erste Block also auch 10% und damit die selbe width. Der vierte Block hat die doppelte width wie der zweite Block (20%) und damit die selbe width. Das liegt daran dass die

breite des Puls mit der Periode wächst und der Dritte und vierte Block die Hälfte der Periodenzeit haben wie der erste und zweite Block. Die Frequenzunterschiede sind auch auf die unterschiedlichen Periodenzeiten zurückzuführen ($freq = \frac{1}{periode}$)

Der Intervall zwischen zwei Spektrallinien sollte nach der Formel 2kHz sein, das lässt sich so ungefähr auch im Spectrum Analyzer ablesen mithilfe von dem Data cursor. Der erste Nullpunkt des Pulses ist abhängig von der Pulse width und der Periodenzeit. So kleiner die Periodenzeit desto größer die Pulsgröße (also desto größer der Nullpunkt), genauso beeinflusst auch die Pulse width die Pulsgröße so kleiner sie ist desto später kommt der Nullpunkt.

Exercise 3: Period signal generation

Um ein bestimmtes Signal mit einem bestimmten Spectrum zu erzeugen, brauchen wir die Grundfrequenz, den ersten Nullpunkt des Spectrums und müssen dann die Periode und die Pulse width berechnen. Auf dem Bild kann man erkennen, dass die Grundfrequenz 10 kHz beträgt.

Dazu habe ich folgendes Matlab Script geschrieben:

```
%the first null in the Envelope is at 80kHz
% f_0 = 2kHz
% T = 1/f_0

T = 1/10000;

% for the Pulse width:
% f = 1/roh
% f = 80kHz
roh = 1/80000;

%period width percent = rho/period *100

w = roh/T * 100
```

Das Script berechnet die Periode als $1e-4$ und die Pulse width als $1.25e-5$, damit die prozentuale Pulse width mit 12,5.

Am Ende muss man noch die Amplitude anpassen, so dass man das richtige Spectrum bekommt. Mit der Amplitude von 0,5 bekommt man ein sehr ähnliches Spectrum:

