

Hausübung 0 – Leon Knauf

■ test_fourier.m

Es müssen lediglich die Variablen **Beispiel** und **Option** angepasst werden, um alle geforderten Plots generieren zu können. **Option** wird dabei nur in der aufgerufenen function ausgewertet.

Die Anzahl an dargestellten Linien in dem ersten Plot ist auf einen Maximalwert von 10 begrenzt:

```
126 % Anzahl an Kurven wenn nötig begrenzen
127 y = y(1:min(10, size(y,1)),:);
128
```

■ fourier_series_fun_02.m

Hier werden alle Daten generiert, um die geforderten Plots darstellen zu können.

Mittels der Variablen **loesungsweg** kann bei Übergabe von **opt=2** oder **opt=3** zwischen 2 Berechnungsverfahren gewählt werden. Wird die Variable mit 1 beschrieben, wird ein iterativer Ansatz mittels for-Schleifen genutzt. Bei **loesungsweg=2** werden die Werte rein durch Matrizenoperation generiert.

Außerdem wird ein rekursiver Ansatz verfolgt, um gleiche Berechnungen nur einmal implementieren zu müssen. So wird beispielsweise bei Aufruf der Funktion mit **opt=1** zunächst die Funktion intern mit **opt=2** aufgerufen, um alle Harmonischen zu bestimmen, aus denen wiederum die Funktionswerte für y berechnet werden:

```
case 1 % nur die Funktion f(t) darstellen

% Alle Harmonischen bestimmen
summs = fourier_series_fun_02(a0,a,b,T,A,t,2);

% Werte für alle Koeffizienten summieren und a0/2 addieren
y = a0/2 + sum(summs,1);
```

Die Funktion wurde über die Aufgabenstellung hinaus um eine vierte Option erweitert, bei der die Funktionswerte mittels der zuvor entwickelten Funktion **fourier_series_fun_01.m** berechnet werden. Diese Option eignet sich, um verschieden Berechnungsverfahren einfach vergleichen zu können.

■ fourier_series_fun_01.m

Diese Funktion wurde auch mit einer **loesungsweg** Variablen versehen, um zwischen der in Teilaufgabe 5 vorgestellten und einer iterativen Lösung wechseln zu können.

```
% Variable zur Auswahl des Lösungswegs
loesungsweg = 1; % 1: eigene Lösung; 2: Lösung aus Teilaufgabe 5
```