Dr. R. Käppeli
M. Sprecher

Musterlösung 9

1. Da

$$(F_N)_{km} = \omega_N^{(k-1)(m-1)}$$

ist F_N offensichtlich symmetrisch. Es gilt

$$\begin{aligned}
(F_N F_N^H)_{km} &= \sum_{l=1}^{N} \omega_N^{(k-1)(l-1)} \omega_N^{-(l-1)(m-1)} && (1)\\
&= \sum_{l=0}^{N-1} \omega_N^{(k-m)l} && (2)\\
&= \begin{cases} N & \text{wenn } k = m \\ 0 & \text{sonst} \end{cases} && (3)
\end{aligned}$$

und deshalb $F_N F_N^H = N I_N$ und $\left(\frac{1}{\sqrt{N}} F_N\right)\left(\frac{1}{\sqrt{N}} F_N\right)^H = I_N$.

2. **a)** Siehe Seite 23 in den Vorlesungsunterlagen

b)
```
function [a] = myfft(x)
%x is a vector of size 2^q
%a is the discrete fourier transform


%length of x
l=length(x);
if l==1
    a=x;
else
    xodd=x(2:2:end);
    xeven=x(1:2:end);
    aodd=myfft(xodd);
    aeven=myfft(xeven);
    a=[aeven;aeven]+exp(-2*pi*1i*linspace(0,1-1/l,l)').*[aodd;aodd];
end
```

Bitte wenden!

```
end


%myfft_test
x=rand(2048,1);
norm(fft(x)-myfft(x))
```

c) Die Matrixmultiplikation benötigt Zeit $(O)(N^2)$. Sei g_N die Laufzeit mithilfe der Gleichung aus a). Es ergibt sich die Rekursionsgleichung $g_N \leq 2g_{N/2} + cN$ mit einer Konstanten $c \in \mathbb{R}$. Wir zeigen nun induktiv, dass $g_{2^q} \leq 2^q(cq + g_1)$. Die Behauptung ist offensichtlich richtig für $q = 0$. Zudem gilt mithilfe der Induktionsvoraussetzung

$$g_{2^{q+1}} \leq 2g_{2^q} + c2^{q+1} \leq 2^{q+1}(cq + g_1) + c2^{q+1} = 2^{q+1}(c(q+1) + g_1).$$

Die Behauptung stimmt also auch für $q + 1$. Die Laufzeit ist also $\mathcal{O}(q2^q) = \mathcal{O}(N \log(N))$.

3. **a)**

```
%ftransf.m
%Anzahl Stuetzstellen
N=30;
suptitle('Funktionen im Frequenzraum');
%Generiere Stuetzstellen
theta=linspace(-1,1-1/2/N,N);

%f_1
ftheta1=cos(10*pi*theta);
a1=fft(ftheta1);
subplot(1,3,1)
plot(abs(a1).^2)
title('cos(10\pix)')

%f_2
ftheta2=ftheta1+cos(5*pi*theta);
a2=fft(ftheta2);
subplot(1,3,2)
plot(abs(a2).^2)
title('cos(10\pix)+cos(5\pix)')
```

Siehe nächstes Blatt!

```
%f_3
ftheta3=exp(-pi*theta.^2);
a3=fft(ftheta3);
subplot(1,3,3)
plot(abs(a3).^2)
title('e^{\pix^2}')

print('-dpdf','ftransf.pdf')
```

b) Es ergibt sich:

$$\boldsymbol{\Delta r}(t) = \sum_{j=1}^{16} \gamma_j \cos(\sqrt{\frac{\lambda_j}{m}} t) \boldsymbol{v}_j.$$

c)

```
% Anfangs-Gelenk Koordinaten
theta = 0.25*pi;
r = [1; %  0;x_1
     1; %  1;y_1
     2; %  2;x_2
     1; %  3;y_2
     3; %  4;x_3
     1; %  5;y_3
     4; %  6;x_4
     1; %  7;y_4
     1; %  8;x_5
     0; %  9;y_5
     2; % 10;x_6
     0; % 11;y_6
     3; % 12;x_7
     0; % 13;y_7
     4; % 14;x_8
     0  % 15;x_8
    ];

% Gleichgewichtsmatrix E
c = cos(theta);
s = sin(theta);
%  (  1, 2, 3, 4, 5, 6, 7, 8, 9,10,11,12,13,14,15,16,17,18)
E = [-1, 0, 0, c, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0;  %  1
      0, 0, 0, s, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0;  %  2
      1,-1, 0, 0, 0, c, 0,-c, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0;  %  3
      0, 0, 0, 0, 0, s, 1, s, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0;  %  4
      0, 1,-1, 0, 0, 0, 0, 0, c, 0,-c, 0, 0, 0, 0, 0, 0, 0;  %  5
```

Bitte wenden!

```
      0, 0, 0, 0, 0, 0, 0, 0, s, 1, s, 0, 0, 0, 0, 0, 0, 0;  %  6
      0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,-c, 0, 0, 0, 0, 0;  %  7
      0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, s, 0, 0, 0, 0, 0;  %  8
      0, 0, 0, 0, 0,-c, 0, 0, 0, 0, 0, 0, 0, 1,-1, 0, 0, 0;  %  9
      0, 0, 0, 0,-1,-s, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0;  % 10
      0, 0, 0, 0, 0, 0, 0, 0,-c, 0, 0, 0, 0, 0, 1,-1, 0, 0;  % 11
      0, 0, 0, 0, 0, 0,-1, 0,-s, 0, 0, 0, 0, 0, 0, 0, 0, 0;  % 12
      0, 0, 0, 0, 0, 0, 0, c, 0, 0, 0, 0, 0, 0, 0, 1,-1, 0;  % 13
      0, 0, 0, 0, 0, 0, 0,-s, 0,-1, 0, 0, 0, 0, 0, 0, 0, 0;  % 14
      0, 0, 0, 0, 0, 0, 0, 0, 0, 0, c, 0, 0, 0, 0, 0, 1,-1;  % 15
      0, 0, 0, 0, 0, 0, 0, 0, 0, 0,-s,-1, 0, 0, 0, 0, 0, 0;  % 16
    ];

% aeussere Kraefte p
m = 1;  % Gelenk Masse
g = 10 ;% Schwerkraft
p = [  0;   %  1
     -m*g;  %  2
       0;   %  3
     -m*g;  %  4
       0;   %  5
     -m*g;  %  6
       0;   %  7
     -m*g;  %  8
       0;   %  9
     -m*g;  % 10
       0;   % 11
     -m*g;  % 12
       0;   % 13
     -m*g;  % 14
       0;   % 15
     -m*g;  % 16
    ];

% Diagonal Matrix D^-1 = Di
Di = diag([1/norm([r( 1),r( 2)] - [r( 3),r( 4)]) % Stab  1
           1/norm([r( 3),r( 4)] - [r( 5),r( 6)]) % Stab  2
           1/norm([r( 5),r( 6)] - [r( 7),r( 8)]) % Stab  3
           1/norm([   0,   0] -   [r( 1),r( 2)]) % Stab  4
           1/norm([r( 9),r(10)] - [r( 1),r( 2)]) % Stab  5
           1/norm([r( 9),r(10)] - [r( 3),r( 4)]) % Stab  6
           1/norm([r(11),r(12)] - [r( 3),r( 4)]) % Stab  7
```

Siehe nächstes Blatt!

```
            1/norm([r( 3),r( 4)] - [r(13),r(14)]) % Stab  8
            1/norm([r(11),r(12)] - [r( 5),r( 6)]) % Stab  9
            1/norm([r(13),r(14)] - [r( 5),r( 6)]) % Stab 10
            1/norm([r( 5),r( 6)] - [r(15),r(16)]) % Stab 11
            1/norm([r(15),r(16)] - [r( 7),r( 8)]) % Stab 12
            1/norm([   5,   0] -   [r( 7),r( 8)]) % Stab 13
            1/norm([   0,   0] -   [r( 9),r(10)]) % Stab 14
            1/norm([r( 9),r(10)] - [r(11),r(12)]) % Stab 15
            1/norm([r(11),r(12)] - [r(13),r(14)]) % Stab 16
            1/norm([r(13),r(14)] - [r(15),r(16)]) % Stab 17
            1/norm([   5,   0] -   [r(15),r(16)]) % Stab 18
           ]);

% Matrix A austellen
eta = 2500; % Elastizitaetsmodule
% Stelle hier die Matrix A auf
% A
A = eta*E*Di*E';


%Eigenwerte w und Eigenvektoren v der Matrix zum Vergleich
[V,D] = eig(A);
w=diag(D);

%Eigenfrequenzen
freq=sqrt(w/m)/2/pi


% plot der x,y Koordinaten Bewegungen des 1. Gelenks
plot(time,x_01,time,y_01)
xlabel('Time');
ylabel('Auslenkung');
legend('x_1','y_1');
print('-dpdf','BewGel.pdf')

%Fourier transforamation fuer die x,y Koordinaten des 1. Gelenks
ax=fft(x_01);
ay=fft(y_01);

%Zeitintervall der Messung: [0,T] mit T = 20 s
T = 20;
```

Bitte wenden!

```
%Anzahl Messunkte
N=4096;


%Energiespektrum
freq=(1:N)/T;
figure;
loglog(freq,abs(ax).^2,freq,abs(ay).^2)
title('Energiespektrum')
xlabel('Frequenz')
ylabel('Quadrat des Betrags des entsprechenden Fourierkoeffizienten'
print('-dpdf','Energiespektrum.pdf')
freq(find(abs(ax).^2+abs(ay).^2>10^2))
```

Das Energiespektrum ist in Abbildung 2 zu sehen. Daraus und mit der Beziehung für die Eigenfrequenz $\omega = \frac{\sqrt{\lambda}}{2\pi}$ sehen wir, dass das gemessene Signal aus den Eigenschwingungen mit den Frequenzen $\omega \approx 1.5096$, $\omega \approx 3.1108$ und $\omega \approx 15.0930$ zusammengesetzt ist.

4. **a)** Es gilt

$$\begin{aligned} \alpha_k &= \int_0^1 e^{-2\pi ikt} f(t)dt && (4)\\ &= \int_a^b e^{-2\pi ikt} dt && (5)\\ &= \begin{cases} (b-a) & k=0 \\ \frac{-1}{2\pi ik} e^{-2\pi ikt}\big|_a^b & k \neq 0 \end{cases}. && (6) \end{aligned}$$

Zudem ist

$$\begin{aligned} \frac{-1}{2\pi ik} e^{-2\pi ikt}\big|_a^b &= \frac{1}{k\pi} e^{-\pi ik(a+b)} \frac{e^{\pi ik(b-a)} - e^{-\pi ik(b-a)}}{2i} && (7)\\ &= \frac{1}{k\pi} e^{-\pi ik(a+b)} \sin(\pi k(b-a)). && (8) \end{aligned}$$

b)

Das Maximum der Approximationen ist jeweils

Siehe nächstes Blatt!

n	$\|f_n\|_\infty$
5	1.0942
25	1.0897
100	1.0895
500	1.0892

Es fällt auf das $\|f_n\|_\infty$ nicht gegen $\|f\|_\infty = 1$ sondern gegen einen anderen Wert zu konvergieren scheint. Es treten sogennante Überschwingunen auf. Dieses Verhalten der Approximationsfunktionen wird Gibbssches Phänomen genannt.

c) Die Funktion g_n ist die trigonometrische Interpolation von f.

```
function [a] = trigint(y)
%Trigonometrische Interpolation

m=max(size(y));
n=(m-1)/2;
%Inverse Fouriertransform
yt=ifft(y);

%Reorder vector elements
a=zeros(1,m);
a(1:n+1)=yt(n+1:-1:1);
a(n+2:m)=yt(m:-1:n+2);

end


f=@(x) 1-(x<.25)-(x>.75);


%Es werden die Approximation T_k(1),...,T_k(m) berechnet
k=[5 25 100 500];
m=4;

%Gitterpunkte an denen die Funktionen ausgewertet werden
nx=10000;
x=linspace(0,1,nx);

%Werte der Approximationen an Gitterpunkten
y=zeros(m,nx);

%Werte der Basisfunktionen an Gitterpunkten
W=zeros(2*k(m)+1,nx);
```

Bitte wenden!

```
kmax=k(m);
for j=-kmax:kmax
    W(j+kmax+1,:)=exp(2*pi*1i*j*x);
end

%trigonometrische Interpolation
for i=1:m
    yk=f(linspace(0,1-1/(2*k(i)+1),2*k(i)+1));
    fhat=trigint(yk);
    y(i,:)=fhat*W(kmax+1-k(i):kmax+1+k(i),:);
end
y=real(y);


%exakte Werte
fx=f(x);

%Fehler berechnen
err=max(y,[],2)

%Funktionen plotten
figure(2);
plot(x,fx,x,y(1,:),x,y(2,:),x,y(3,:),x,y(4,:));
legend('f',['T_{' num2str(k(1)) '}'],['T_{' num2str(k(2)) '}'],...
    ['T_{' num2str(k(3)) '}'],['T_{' num2str(k(4)) '}']);
    print('-dpdf','trigint_treppe.pdf');
```

Es ergiebt sich ein änliches Bild wie in b). Das Maximum der Approximationen ist jeweils

n	$\|f_n\|_\infty$
5	1.1873
25	1.1507
100	1.1436
500	1.1404

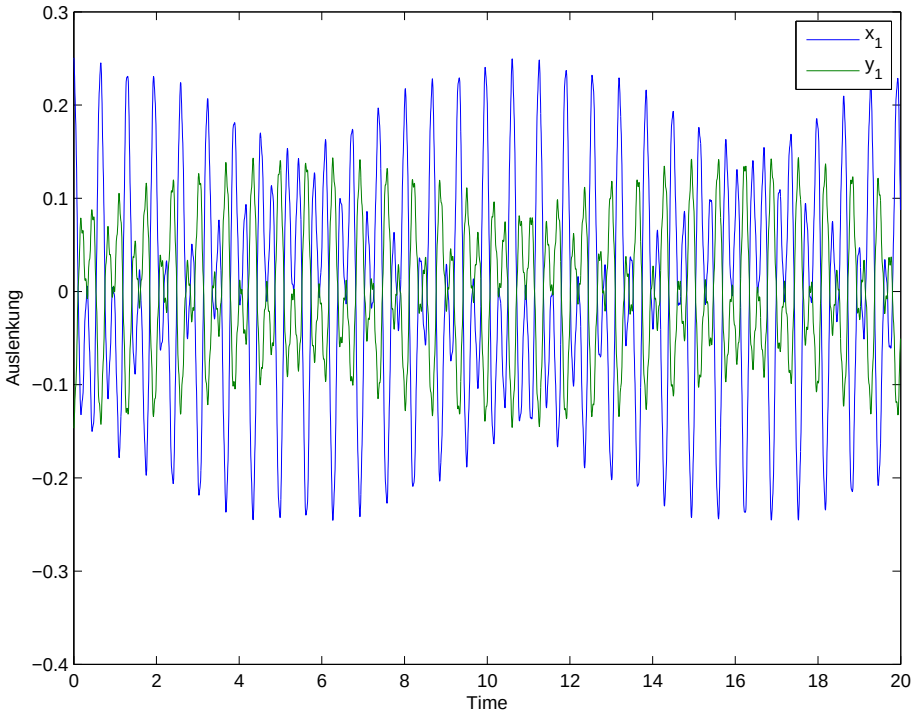


Abbildung 1: Bewegungen des 1. Gelenks.

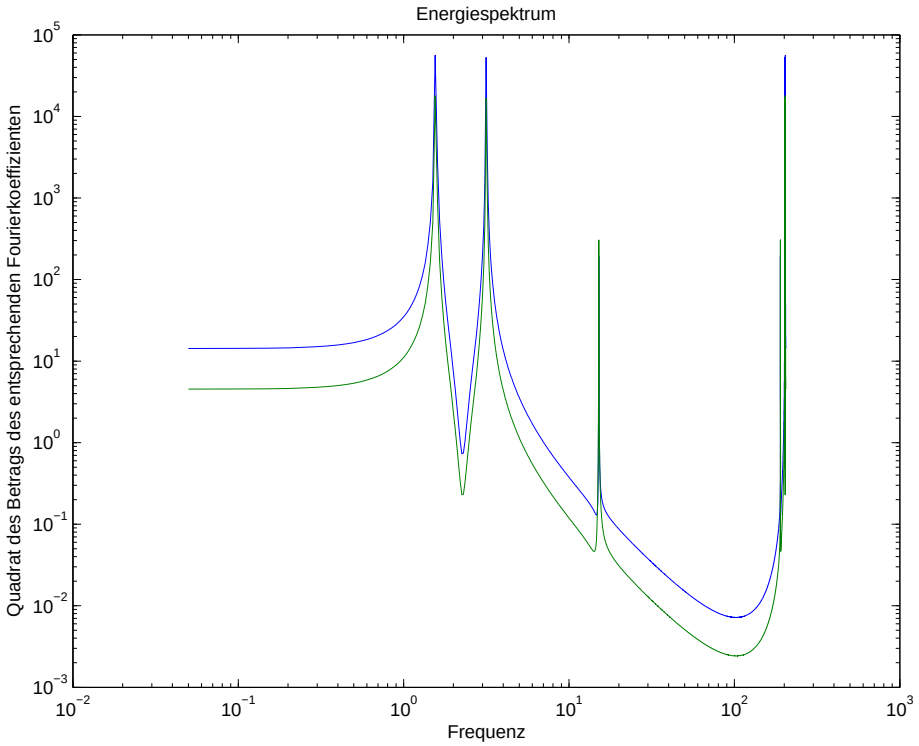
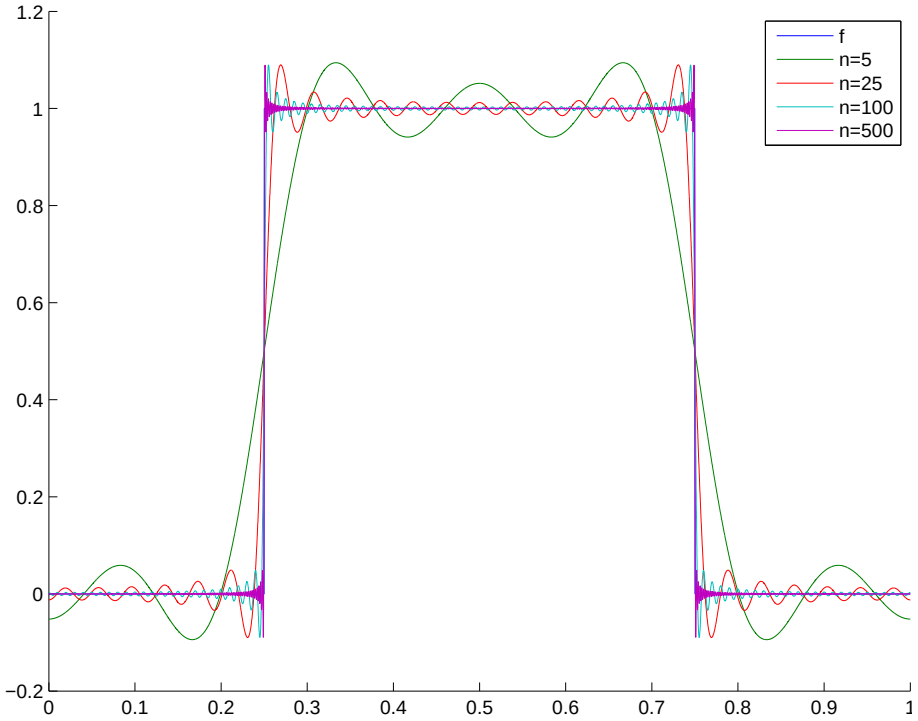
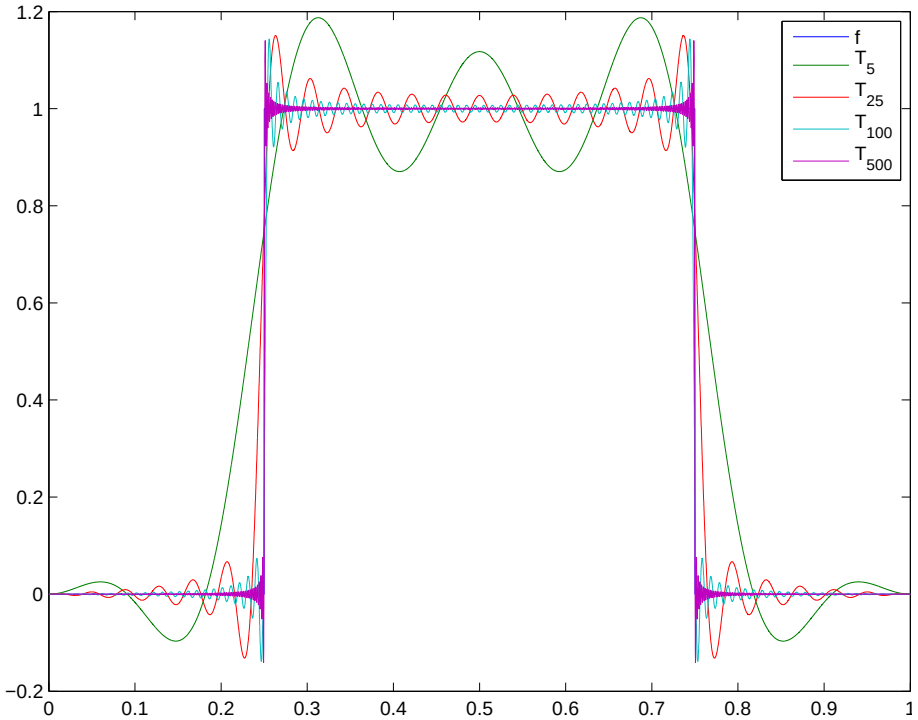


Abbildung 2: Energiespektrum des gemessenen Signals.


f
n=5
n=25
n=100
n=500
1.2
1
0.8
0.6
0.4
0.2
0
−0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


f
T_5
T_{25}
T_{100}
T_{500}
1.2
1
0.8
0.6
0.4
0.2
0
−0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1