Dr. R. Käppeli
M. Sprecher

Musterlösung 10

1. Für $l \in \{0, \ldots, n-1\}$ gilt

$$
\begin{aligned}
(C\vec{\omega}_n^k)_l &= \sum_{j=0}^{n-1} c_{l-j}\omega_n^{kj} && (1)\\
&= \omega_n^{kl}\sum_{j=0}^{n-1} c_{l-j}\omega_n^{k(j-l)} && (2)\\
&= \omega_n^{kl}\sum_{m=l-n}^{l-1} c_m\omega_n^{-km} && (3)\\
&= \omega_n^{kl}\left(\sum_{m=0}^{l-1} c_m\omega_n^{-km} + \sum_{m=l-n}^{-1} c_m\omega_n^{-km}\right) && (4)\\
&= \omega_n^{kl}\left(\sum_{m=0}^{l-1} c_m\omega_n^{-km} + \sum_{m=l}^{n-1} c_m\omega_n^{-km}\right) && (5)\\
&= \omega_n^{kl}\sum_{m=0}^{n-1} c_m\omega_n^{-km}. && (6)
\end{aligned}
$$

2. Wir verwenden fft an um die Konvolution effizient zu berechnen.

```
function [y] = circtimesx(c,x)
y=ifft(fft(c).*fft(x));
end
```

3. **a)** • Direkt mit Definition

```
function [c] = conv(a,b)
%berechnet diskrete Faltung zweier endlicher Vektoren
na=max(size(a));
```

Bitte wenden!

```
nb=max(size(b));
a=[a zeros(1,nb-1)];
b=[b zeros(1,na-1)];
c=ones(1,na+nb-1);
for i=1:na+nb-1
    c(i)=sum(a(1:i).*b(i:-1:1));
end

end
```

- Mittels schneller Fouriertransformation

```
function [c]=convwithfft(a,b)
%berechnet diskrete Faltung zweier endlicher Vektoren mit fft
na=max(size(a));
nb=max(size(b));
a=[a zeros(1,nb-1)];
b=[b zeros(1,na-1)];

c=ifft(fft(a).*fft(b));
end
```

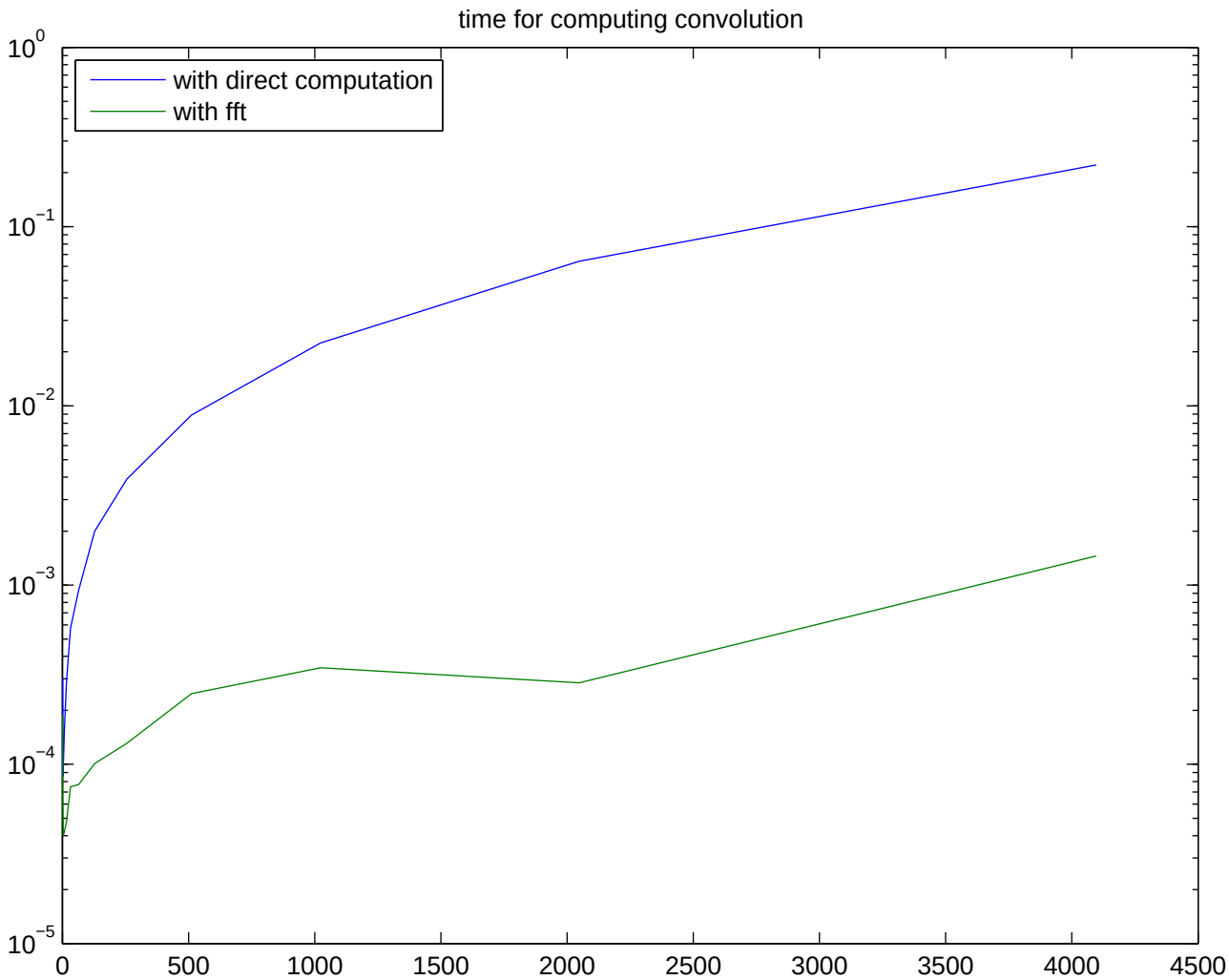
b)

```
%Werte der verschiedenen n's
n=2.^(0:12);


%Vektor mit Zeiten
t=zeros(2,0);


for i=n
    a=1:i;
    b=i:-1:1;
    tic
    c1=conv(a,b);
    t(1,end+1)=toc;
    tic
    c2=convwithfft(a,b);
    t(2,end)=toc;
    norm(c1-c2)
end
%Plot
semilogy(n,t(1,:),n,t(2,:))
```

Siehe nächstes Blatt!




```
title('time for computing convolution');
legend('with direct computation','with fft','Location','NorthWest')
print('-dpdf','fft_vs_naive')
```

4. **a)**

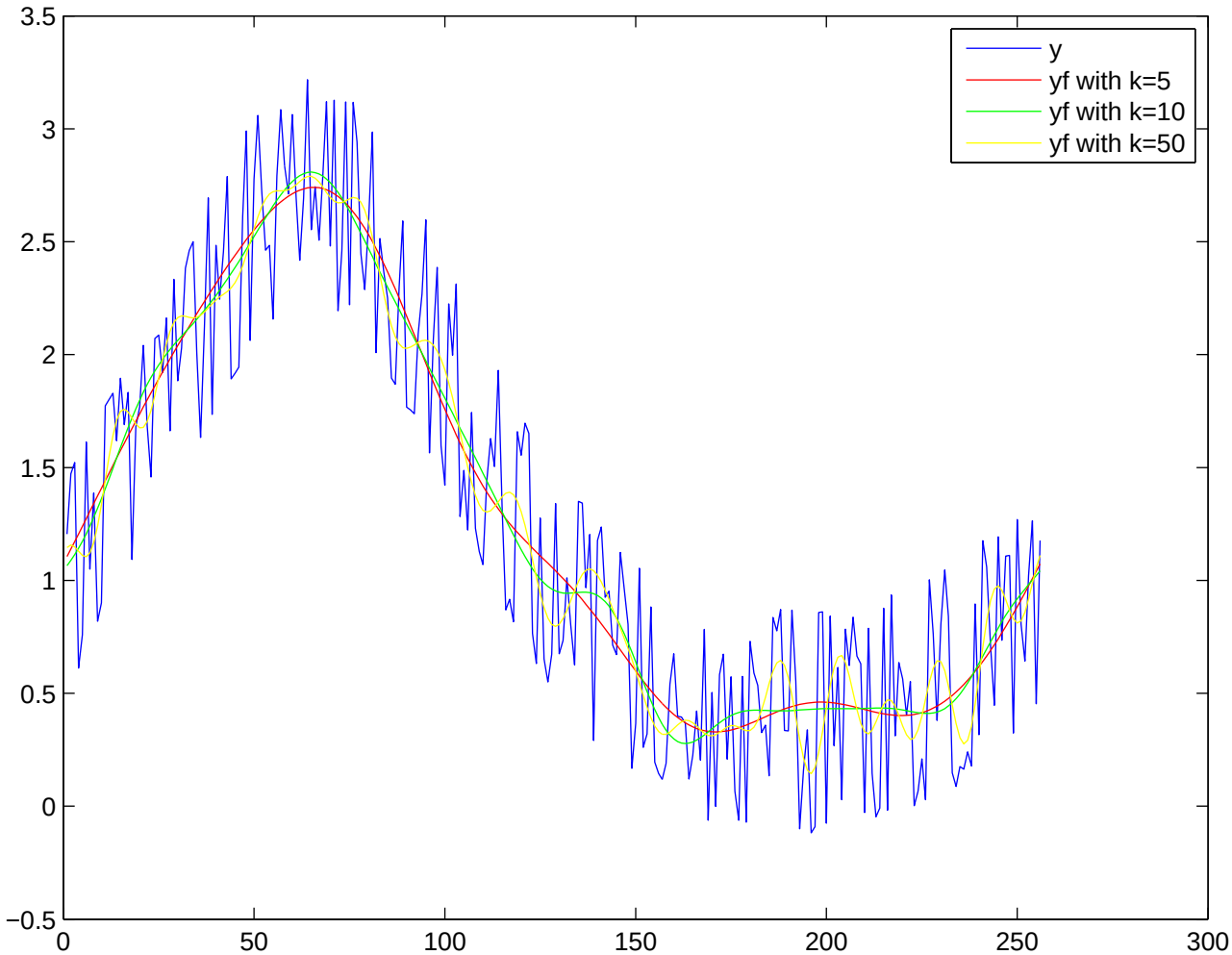
```
function [ynew] = freqfilter(y,k)

l = length(y);
c = fft(y);
clow = c;
clow(k+2:l-k) = 0.;
ynew  = real(ifft(clow));
```

b) `%Freqsignal.m`

Bitte wenden!

```
plot(y);
hold on;
k=[5,10,20]; %Intervallaengen die rausgefiltered werden
C=['r','g','y'];% Colors
for i=1:3
ffy=freqfilter(y,k(i));
plot(ffy,C(i));
hold on;
end
legend('y',['yf with k=' num2str(K(1))],['yf with k=' num2str(K(2))]
print('-dpdf','filterfreq.pdf')
```



c)
```
function [ynew] = myfilter(y,j)
N=max(size(y));
aj=fft([.5;.25;zeros(N-3,1);0.25]).^j;
ynew=ifft(aj.*fft(y));
```

Siehe nächstes Blatt!

```
end
```

d)
```
%Signal.m
plot(y);
hold on;
j=[5,10,100];
C=['y','g','r'];% colors
ynew=y;
for i=1:3
yj=myfilter(ynew,j(i));
plot(yj,C(i));
hold on;
end
legend('y',['myfilter(y,' num2str(j(1)) ')'],['myfilter(y,' num2str(j
print('-dpdf','filterconv.pdf')
```