FFT and IFFT in Maple and Matlab

Nasser M. Abbasi

August 2005 Compiled on September 9, 2023 at 10:00am

This example show how to do FFT on a continouse time function, then do an IFFT to recover the original function. This is done in Matlab and Maple

1 Matlab

```
clear all;
t = linspace(-pi,pi,100);
dt = t(2)-t(1);
fs = 1/dt;
w = linspace(0,fs,100);

y = sin(t) + 2*sin(3*t);
g = exp(-w) .* fft(y);
h = ifft(g);
plot(real(h));
```

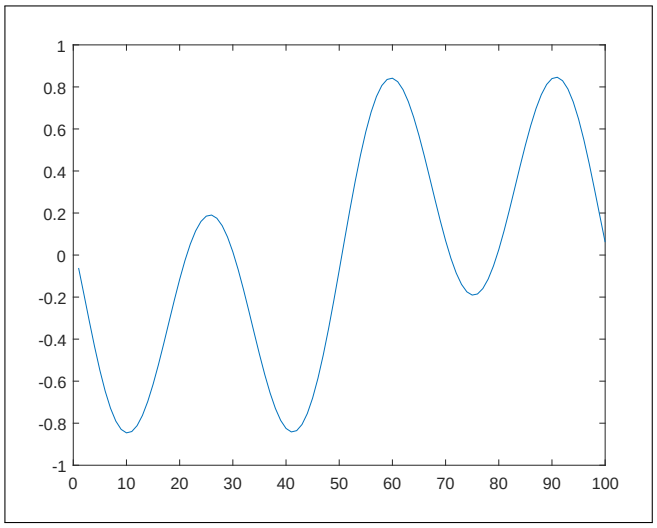

Figure 1: Output of the above

2 Maple

```
#showing how to do FFT and IFT in maple 10. Nasser M. Abbasi.
restart;
with(inttrans):
with(LinearAlgebra):
with(plots):
N:=100:

#the function to fft
y:= t->sin(t) + 2*sin(3*t):

linspace:=proc(fromP,toP,n)
          local incr,data,i,T,L;
          incr:=(toP-fromP)/(n-1);
          L:=Vector(1..n,[]);
          T:=fromP;
          for i from 1 to n do
              L[i]:=T;
              T:=T+incr;
          end do:
          return(L);
end proc:

data:=Vector(1..N,[]):
g:=Vector(1..N,[]):
T:=linspace(-Pi,Pi,N):

for i from 1 to N do
    data[i]:=evalf(y(T[i]));
end do:

Y:=DiscreteTransforms:-FourierTransform( data,algorithm=DFT,padding=0 ):
adj:=evalf(sqrt(1/N)): #adjust as maple has normalization factor
Y:=Y/adj:

dt:=T[2]-T[1]:
fs:=1/dt:
w:=evalf(linspace(0,fs,N)):
s:=[seq(exp(-w[i]),i=1..N)]:
g:=Vector(N,zip((x,y)->x*y,s,convert(Y,list))):
h:=DiscreteTransforms:-InverseFourierTransform( g ):
h:=h*sqrt(1/N):#adjust the IFFT due to maple normalization again

listplot(map(Re,h),color=red,labels=["t","y(t)"],title="result of IFFT");
```

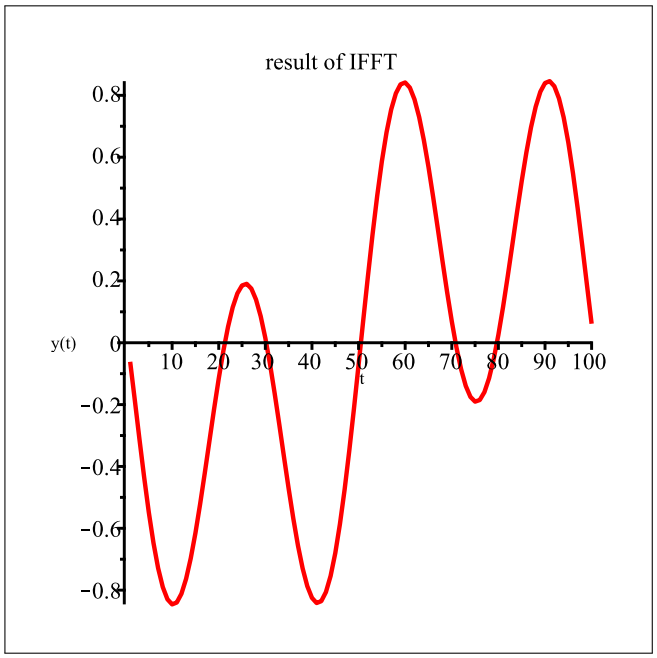

Figure 2: Output of the above

3 Source code download

1. Matlab `fft_example.m`
2. Maple maple.txt