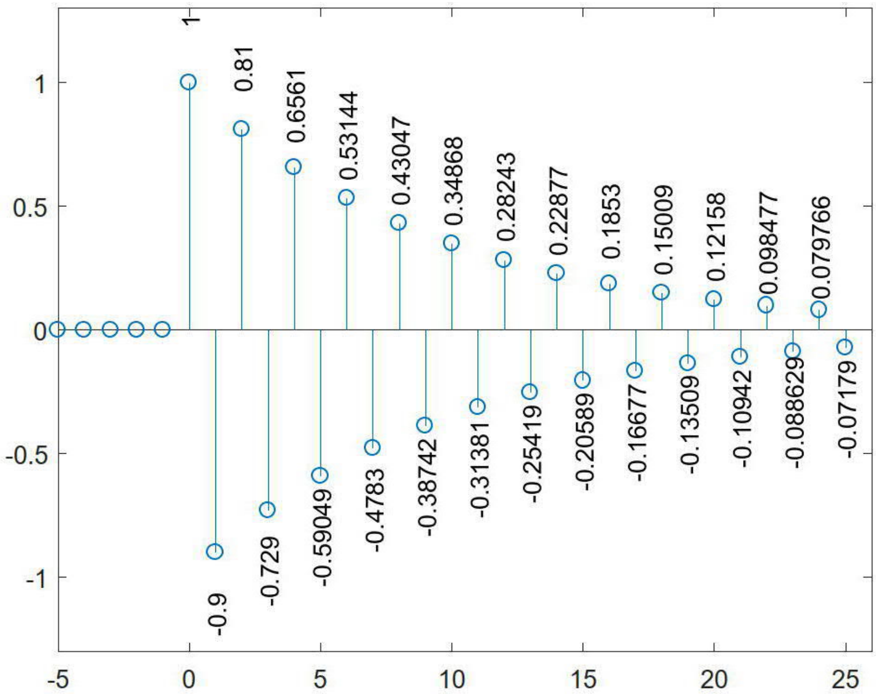
SGN-11007 Introduction to Signal Processing,
Exercise 3, Fall 2020

Task 1. *(Pen & paper)* Calculate the DTFT of the infinitely long signal shown in the figure.



Task 2. *(Pen & paper)* Calculate manually the discrete Fourier transform of the vector $x(n) = (5, 1, -1, 0)^T$.

Task 3. *(Pen & paper)* Calculate the DFT of the sequence $x(n) = (-1, 3, 1, 0)$ using the FFT algorithm. You can skip part of the calculations by utilizing this information: the DFT of the sequence $(-1, 1)$ is $(0, -2)$ and the DFT of $(3, 0)$ is $(3, 3)$.

Task 4. *(Matlab)* On p. 45 of the lecture handout the system $y(n) = 1.1y(n-1)+x(n)$ is stated not to be stable because for the input $u(n)$ the output grows without bound. Try this with Matlab as follows. First, generate the $u(n)$ signal as in Exercise 2, Task 5(b) (however, make the part consisting of ones a bit longer).

Matlab filters the signal x with the command

```
y=filter(b,a,x);
```

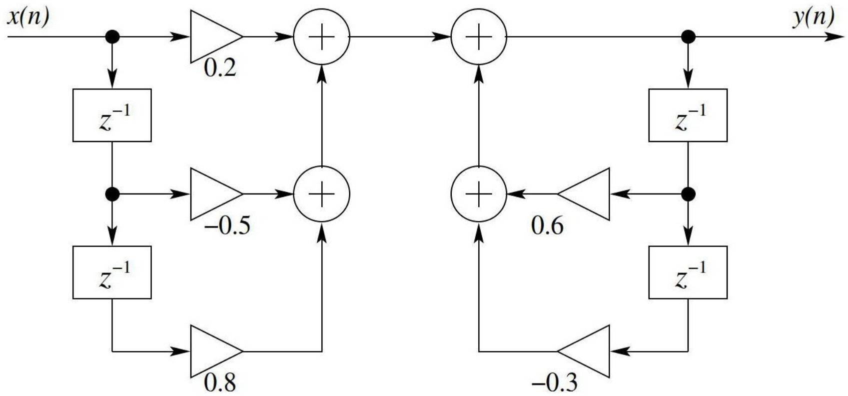
where the vector b contains the feedforward filter coefficients and the vector a feedback filter coefficients. Note the comment on p. 50 concerning the coefficients in Matlab.

Filter $u(n)$ with this system using the command `filter`. Plot the output using the command `stem`.

Task 5. *(Matlab)* Generate the impulse signal $\delta(n)$ as follows: `delta=[1,zeros(1,127)];` Filter (`filter`) this with the following systems and plot the outputs:

(a) $y(n) = 0.75y(n-1) + x(n)$,

(b) system in the figure below



(c) $y(n) = x(n) + 0.5x(n-1) + 1.25x(n-2) - 0.8y(n-1) + 0.8y(n-2)$.

One of these three systems is not stable. Can you tell by studying the impulse responses which of them it is?

Task 6. *(Matlab)* Plot the impulse responses in the previous task using Matlab's command `impz`. The input parameters are vectors `b` and `a`.

Task 7. *(Matlab)* Generate a one second long signal having frequency 2000 Hz with sampling rate 16000 Hz. Calculate the DFT of the signal using Matlab command `fft` and plot the graph of its absolute values. (`help fft, help plot, help abs`). The figure should have a clear spike in two positions on the horizontal axis (corresponding to the frequency 2000 Hz).

Task 8. *(Matlab)* We compare the computation times of DFT and FFT in this and the following task.

(a) Implement the function `dft(x)`, which works like the `fft(x)` but calculates the result directly by matrix multiplication. There are two steps in the implementation: (1) construct the DFT matrix and (2) left-multiply the input vector `x` by the matrix. The DFT matrix can be created with the command

```
F=exp(-2*pi*1i*(0:N-1)'*(0:N-1)/N);
```

The variable `N` is the number of elements in the vector `x`, which you can obtain with the `length` function.

(b) Check that the `dft` and `fft` give the same result e.g. for the vector $x = [1, 2, 3, 4]$.

Task 9. *(Matlab)* Now we are ready to compare the computation times.

(a) First, test both functions with a random vector of length 1024 `x=rand(1024,1)`. The execution time is obtained by using `tic/toc` pair as follows:

```
tic(); % Starts a stopwatch timer
X=dft(x); % DFT computation
elapsed_time=toc(); % Reads the elapsed time from the timer
```

(b) Put the calculation inside a `for`-loop and do the same calculation 100 times to get a more accurate estimate.

(c) Furthermore, put the 100 times calculation inside a second loop where you perform tests for lengths $N = 32, 64, 128, 256, 512, 1024$.

(d) Plot the time versus length graphs for FFT and DFT.

Task 10. *(Matlab)* In the course Moodle `Ex3_Task10.mat` (`Ex_3.zip`) is a corrupted version of Matlab's test signal `handel`. Your task is to find the impulse response of the distortion process. Read the file into Matlab. It contains the variables `x` (original) and `y` (distorted). The distorted signal is obtained by convolving impulse response $h(n)$ with the original signal

$$y(n) = h(n) * x(n).$$

Now you have the vectors $x(n)$ and $y(n)$ in Matlab. Solve $h(n)$ with FFT and plot the first 10 terms.