

Guidelines

- Write your name and matriculation number on each sheet of paper.
- Only clearly readable exercise-elaborations are evaluated.
- Results have to be provided together with an evident way of calculation.
- Keep textual answers short and concise. Lengthy or vague statements won't gain points.

Exercise 1.1 (0.5 points)

Consider the **time-invariant** system T in Figure 1.1. From the input sequences $x_{1,n}$, $x_{2,n}$ and $x_{3,n}$ to the system T result the corresponding output sequences $y_{1,n}$, $y_{2,n}$ and $y_{3,n}$, as shown in the figure.

1. (a) Is the system T linear?
- (b) From the given input/output relations, can you determine which output sequence y_n results from an input sequence $x_n = \delta_n$? If not, which input/output relations let you do so?
- (c) Find all possible input sequences x_n , for which the system responses of the system T can be determined from the given information only.

In the following, consider the **linear** system L in Figure 1.2. From the input sequences $x_{1,n}$, $x_{2,n}$ and $x_{3,n}$ to the system L result the corresponding output sequences $y_{1,n}$, $y_{2,n}$ and $y_{3,n}$, as shown in the figure.

2. (a) Is the system L time-invariant?
- (b) What output sequence y_n results from an input sequence $x_n = \delta_n$? Calculate the impulse response for the time instance $n = 0$, $L[\delta_n] = h_{0,n}$.

Terms: system properties, linearity, time invariance.

Exercise 1.2 (0.5 points)

Consider a causal linear system whose transfer function is described by the polynomial¹

$$H(q^{-1}) = \sum_{i=0}^{n_H-1} h_i q^{-i}.$$

¹The linear operator q^{-1} denotes the unit delay defined as $q^{-1}[x_k] = x_k q^{-1} = x_{k-1}$.

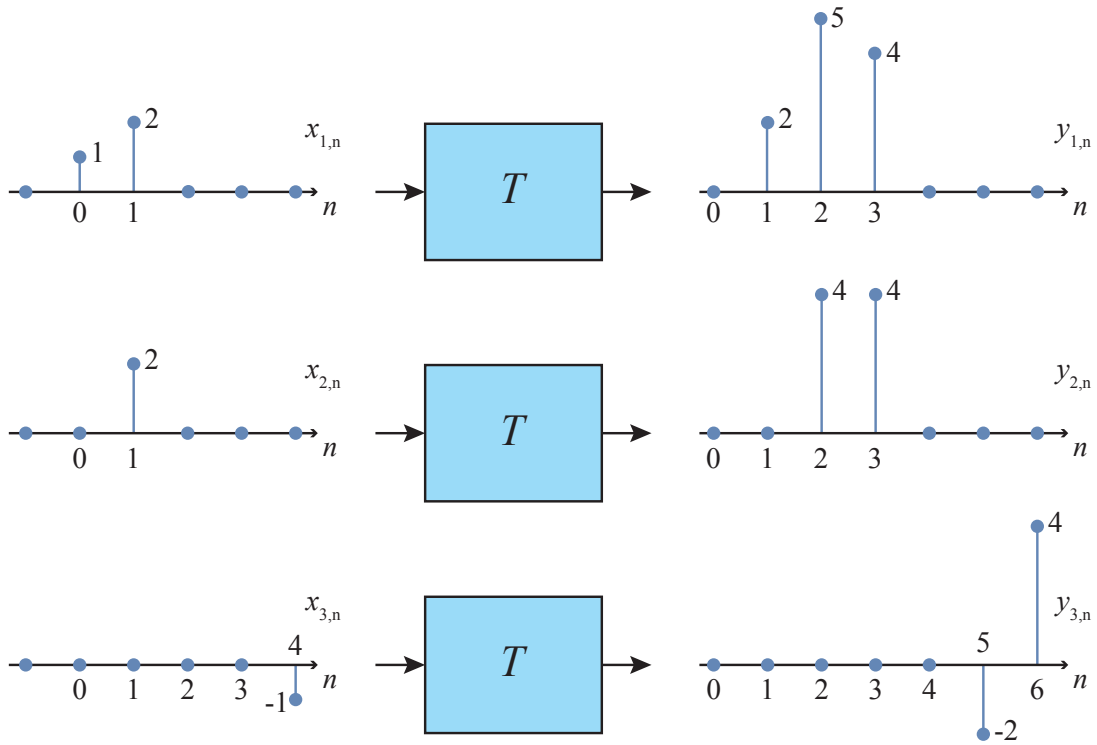


Figure 1.1: time-invariant system T .

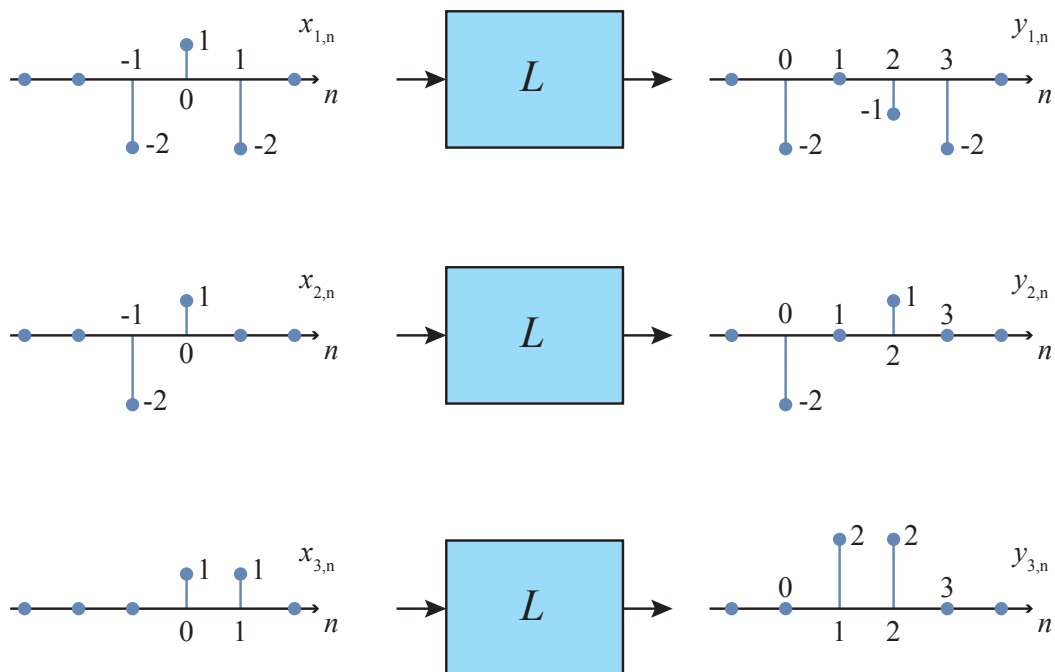


Figure 1.2: Linear system L .

An input sequence x_k that passes through this system shall be equalized by

$$G(q^{-1}) = \sum_{i=0}^{n_G-1} g_i q^{-i},$$

where n_G equalizer coefficients g_i are used. The setup is depicted in Figure 1.3.



Figure 1.3: Basic setup of linear system and its equalizer.

Assuming that we know $H(q^{-1})$, our objective is to determine $G(q^{-1})$. In order to obtain an undistorted signal after D time instances, i.e., $y_{k+D} = x_k$, we have to determine the equalizer coefficients that satisfy $H(q^{-1})G(q^{-1}) = q^{-D}$. Assume finite order polynomials ($n_G, n_H < \infty$).

1. Write the problem in matrix-vector form — how many equations and how many unknowns do we have? How many solutions does the problem have in general, and why?
Hint: you may assume fixed transfer function lengths at first, e.g.: $n_H = 5, n_G = 2$ and also $n_H = 2, n_G = 5$. Then, try to find the general form.
2. Let us now separate the polynomials according to

$$H(q^{-1}) = H^{(1)}(q^{-1}) + H^{(2)}(q^{-1}) = \sum_{i=0}^{n-1} h_i q^{-i} + \sum_{i=n}^{2n-1} h_i q^{-i},$$

$$G(q^{-1}) = G^{(1)}(q^{-1}) + G^{(2)}(q^{-1}) = \sum_{i=0}^{n-1} g_i q^{-i} + \sum_{i=n}^{2n-1} g_i q^{-i},$$

where $n_H = n_G = 2n$ (n_H is divisible by 2). Utilizing the separated polynomials, we construct the setup depicted in Figure 1.4.

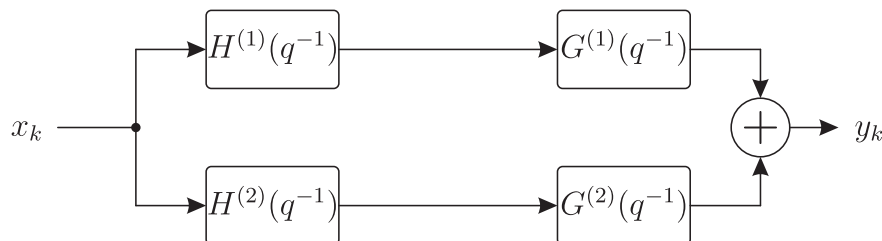


Figure 1.4: Separated setup of linear system and equalizer.

Note that the overall transfer function is different from that in Figure 1.3. Write the problem of determining the equalizer coefficients in matrix-vector form (as a single system of linear equations). How many equations and how many unknowns do we have in general? For which value(s) of n_H does a solution exist?

Utilizing the separated blocks ($H^{(1)}(q^{-1})$, $H^{(2)}(q^{-1})$, $G^{(1)}(q^{-1})$, $G^{(2)}(q^{-1})$), draw the setup that leads to the same overall transfer function as the setup in Figure 1.3.

3. Let us delay the upper branch by n time instances as depicted in Figure 1.5.

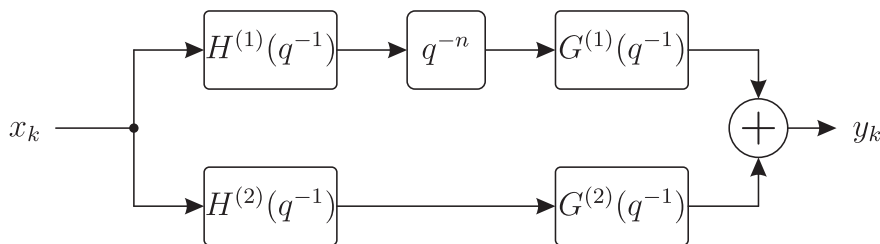


Figure 1.5: Separated setup with delay in branch 1.

How does the new matrix-vector form of the problem look like? How many viable (useful) equations does the system feature? How many solutions does the problem have in general, and why?

4. Now consider the setup in Figure 1.4 with

$$\begin{aligned} H^{(1)}(q^{-1}) &= 1 + q^{-1}, \\ H^{(2)}(q^{-1}) &= -1 + q^{-2}. \end{aligned}$$

Note that these polynomials describe distinct transfer functions and were not obtained by separating a larger transfer function. Show that for this specific choice of system polynomials, there exists no (finite length) solution for the zero-forcing equalizer. Why is that?

How about the alternative polynomials

$$\begin{aligned} H^{(1)}(q^{-1}) &= 2 - 3q^{-1} + q^{-2}, \\ H^{(2)}(q^{-1}) &= 12 - 7q^{-1} + q^{-2}? \end{aligned}$$

Terms: linear operators, polynomial description, matrix-vector description, zero-forcing equalizer, Bezout's theorem.

Exercise 1.3 (0.5 points)

Consider the lossless resonant circuit shown in Figure 1.6.

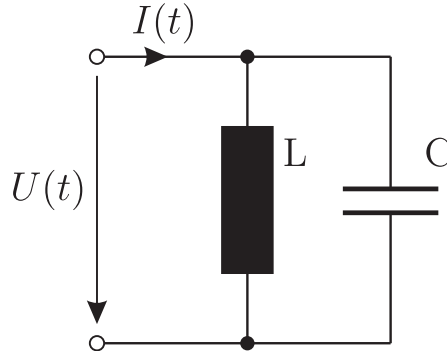


Figure 1.6: Lossless LC resonant circuit.

1. Derive the differential equation connecting the voltage $U(t)$ and the current $I(t)$.
2. Assuming zero initial conditions ($I(0-) = 0 \text{ A}$, $I'(0-) = 0 \text{ A/s}$, $U(0-) = 0 \text{ V}$, $U'(0-) = 0 \text{ V/s}$) compute the corresponding voltage $U(s)$ in the Laplace domain, in terms of the Laplace transform of the current $I(s)$.

Considering that $I(s)$ is the input and $U(s)$ is the output of the system, determine the transfer function $H(s)$.

We now intend to examine the system in case of sampled signals. The bilinear transform is a widely used method to design digital filters from already designed analog filters, the s -domain is mathematically transformed to the z -domain by

$$H(z) = H(s) \Big|_{s=\frac{2}{T} \frac{z-1}{z+1}}.$$

3. Compute the transfer function $H(z)$ of the LC resonant circuit. If you were not able to solve the previous problem, assume $H(s) = \frac{\alpha_1 s}{1 + \alpha_2 s^2}$.
4. Bring the transfer function into its canonical form and determine the companion form of the filter, i.e., determine A , \underline{b} , \underline{c} , d .
5. Are there choices of coefficient a_1 (see canonical form) for which the system is stable? (You may use Matlab to check the necessary condition(s).)

Terms: Laplace transform, transfer function, canonical form, companion form, stability.

MATLAB-Exercise 1.1 (1 point)

In this Matlab exercise, we will investigate Bezout's theorem according to a practical example. You will learn to use some basic functions of Matlab and how to write your own functions. Numerical issues with ill-conditioned matrices will also be investigated.

Consider the setup in Figure 1.7. A signal x_k is transmitted over a single antenna and received on two separate antennas. The respective channels are described by the polynomials $H^{(1)}(q^{-1})$ and $H^{(2)}(q^{-1})$. For each branch, an equalizer is used. The equalized receive signal shall obey $y_k = q^{-D}x_k$.

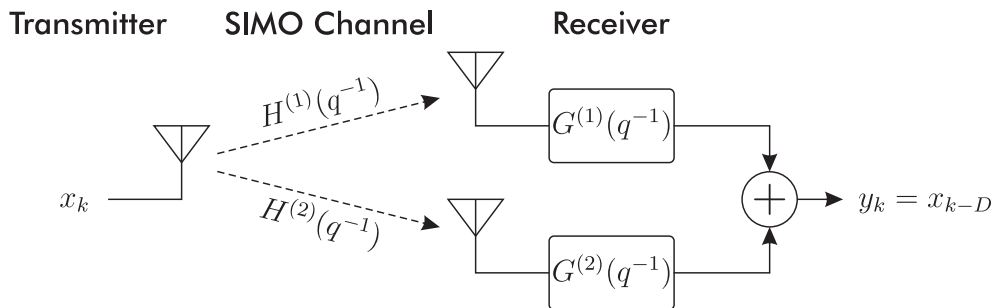


Figure 1.7: SIMO transmission link with equalizer.

1. Write a new function `[X] = myToeplitz(x,N)` with input arguments \underline{x} , N and output X . The function takes vector \underline{x} and arranges it in a Toeplitz matrix X of dimension $(2N - 1) \times N$. In order to meet the size constraint, you have to add zeros to \underline{x} ("zero padding") if `length(x) < N` and remove elements from \underline{x} if `length(x) > N`.
2. From previous measurements, we gained perfect channel knowledge:

$$\begin{aligned} H^{(1)}(q^{-1}) &= 0.8 + 0.5q^{-1}, \\ H^{(2)}(q^{-1}) &= 0.3 + 0.6q^{-1} + 0.1q^{-2}. \end{aligned}$$

Utilizing the matrix-vector notation and Matlab (with `myToeplitz` function), compute the equalizer coefficients that satisfy

$$H^{(1)}(q^{-1})G^{(1)}(q^{-1}) + H^{(2)}(q^{-1})G^{(2)}(q^{-1}) = q^{-D}.$$

To compute coefficient vector \underline{g} in an equation such as² $H\underline{g} = \underline{e}$, use pseudo-inverse $H^\dagger = H^T(HH^T)^{-1}$ so that $\hat{\underline{g}} = H^\dagger \underline{e}$. In our case, the pseudo inverse

²Hint: H is a matrix to perform convolution of the channel transfer functions with the equalizer transfer functions. Note that transfer function $H^{(1)}(q^{-1})$ has less coefficients than $H^{(2)}(q^{-1})$, you have to perform zero padding in the matrix-vector notation to obtain the proper matrix H .

yields the unique minimum norm solution to the underdetermined system of equations $H\hat{g} = \underline{e}$ which in general has an infinite number of solutions. You will hear about pseudo-inverses and minimum norm solution later in this course (Chapter 3 - Approximations).

In order to compute \hat{g} in Matlab, try the following variants:

- use pseudo-inverse function `pinv`
- use left matrix divide `\`
- use `inv` for matrix inversion and `.'` for transposition

Verify your solution by computing

$$\hat{x}_k = (H^{(1)}(q^{-1})G^{(1)}(q^{-1}) + H^{(2)}(q^{-1})G^{(2)}(q^{-1})) x_k$$

in matrix-vector notation with an all-one input vector $\underline{x} = [1, \dots, 1]^T$ of length $M = 10$ and the equalizer coefficients \hat{g} .

Hint: the vector $\hat{e} = H\hat{g}$ has to be arranged in a Toeplitz matrix in order to perform convolution with input signal \underline{x} .

Compute the mean squared error between \underline{x} and its equalized version $\hat{\underline{x}} = [\hat{x}_1, \dots, \hat{x}_{2M-1}]^T$:

$$\text{MSE} = \frac{1}{M} \sum_{i=1}^M |x_i - \hat{x}_{i+D}|^2.$$

It should be numerically zero, i.e., in the order of 10^{-30} .

- Try various values of $D = 0, 1, 2, \dots$ and see if the equalizer still works.
 - Is there a maximal D where Matlab prompts an error? If so, how can you fix this?
3. Let us now investigate numerical issues with poorly conditioned matrices. Assume the channels

$$\begin{aligned} H^{(1)}(q^{-1}) &= 1 + 0.5q^{-1}, \\ H^{(2)}(q^{-1}) &= 1 + (0.5 + \epsilon)q^{-1}. \end{aligned}$$

From Bezout's theorem, we know that an FIR equalizer is only feasible for coprime polynomials, which is not satisfied in case of $\epsilon = 0$. But what happens if ϵ is very small and numerical methods are employed?

Use Matlab to compute $\text{MSE}(\epsilon)$, where ϵ shall range logarithmically (use `logspace`) from 10^{-20} to 10^{-1} .

- Plot MSE vs. ϵ in a double logarithmic plot (use `loglog`).

- Compare all three variants (see 2.) of computing \hat{g} in the same figure.
- Do you observe differences in the numerical fidelity of the equalized signal?

(Note: for small ϵ , Matlab will probably throw warnings at you.)

In order to compute the coefficient vector \hat{g} in 2., matrix H had to be pseudo-inverted. A matrix is said to be poorly- or ill-conditioned if its condition number $\kappa(H)$ is large, which implies that the matrix is almost singular. You can compute the condition number of a matrix in Matlab using `cond`.

- Plot the condition number $\kappa(H)$ vs. ϵ in a double logarithmic plot.
- Can you observe the value of $\kappa(H)$ where the pseudo-inverse of H becomes numerically singular (HH^T becomes rank-deficient)?

Matlab help: Useful functions for this exercise are `zeros`, `ones`, `length`, `circshift`, `logspace`, `loglog`, `max`, `cond`, `inv`, `pinv`, `figure`. By typing `help XXX` into the command window, Matlab prompts the help to function `XXX`. You can concatenate matrices horizontally with `C = [A,B]` and vertically with `C = [A;B]`.