Convolution of Signals

Definitions

- Convolution is a fundamental concept in linear systems theory.
- The convolution theorem states that the response of a system (assuming zero initial conditions or rest) to an arbitrary input signal is formed by the convolution of that input signal with the system's weighting function (impulse response).

The continuous-time convolution of two signals $x_1(t)$ and $x_2(t)$ is defined by the relation:

$$x(t) = x_1(t) * x_2(t) = \int_{-\infty}^{\infty} x_1(\tau)x_2(t-\tau)d\tau \;\; ; \; -\infty < t < \infty$$

In the previous relation, τ is the integration variable and t is a parameter. The duration of a signal $x_i(t)$ is defined by the time moments t_i and T_i such that for any value of t outside the interval $[t_i \; , \; T_i]$ the signal is zero, meaning: $x_i(t) = 0, t \notin [t_i \, , T_i]$

The properties of the convolution integral are:

1. Commutativity:

$$x_1(t) * x_2(t) = x_2(t) * x_1(t)$$

2. Distributivity:

$$x_1(t) * \{x_2(t) + x_3(t)\} = x_1(t) * x_2(t) + x_1(t) * x_3(t)$$

3. Associativity:

$$x_1(t) * \{x_2(t) * x_3(t)\} = \{x_1(t) * x_2(t)\} * x_3(t)$$

4. Duration:

$$x(t) = x_1(t) * x_2(t) = \begin{cases} 0, t \le t_1 + t_2 \\ \displaystyle\int_{t_1+t_2}^{T_1+T_2} x_1(\tau)x_2(t-\tau)d\tau \;\; ; \; t_1 + t_2 \le t \le T_1 + T_2 \\ 0, t \ge T_1 + T_2 \end{cases}$$

5. Time Shift Property

Let $x(t) = x_1(t) * x_2(t)$. The convolution of time-shifted signals is characterized by the following relations:

$$x_1(t-\tau_1) * x_2(t) = x(t-\tau_1)$$

$$x_1(t) * x_2(t-\tau_2) = x(t-\tau_2)$$

$$x_1(t-\tau_1) * x_2(t-\tau_2) = x(t-\tau_1-\tau_2)$$

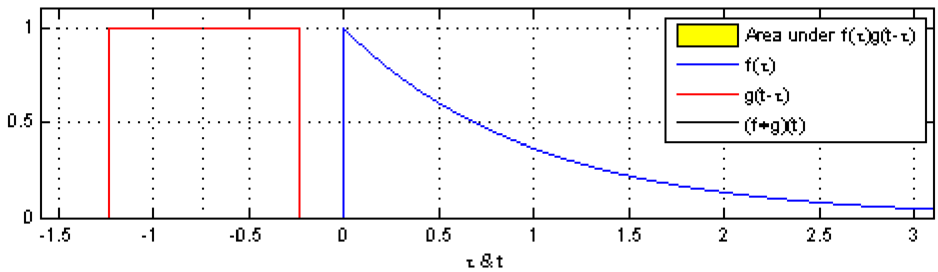
To better understand convolution, three important steps can be defined in calculating the convolution integral:

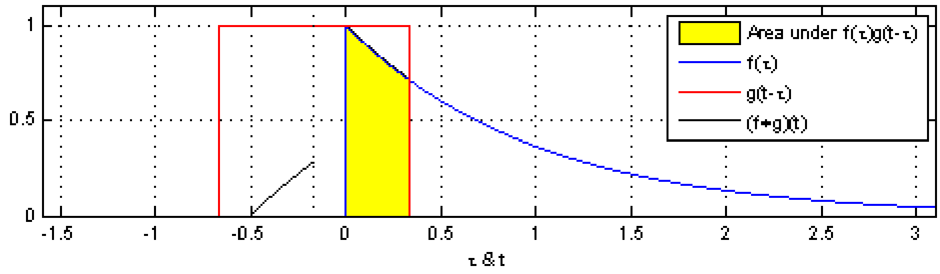
Step 1. Apply the property that characterizes the duration of the convolution to define/identify the intervals where it is zero.

Step 2. Reflect one of the signals across the vertical axis (Oy), i.e., represent one of the signals with respect to the time coordinate $-\tau$.

Step 3. Vary the parameter *t* from $-\infty$ to ∞ => meaning the mirrored (reflected) signal will shift from left to right along the time axis. During this shift, observe the intervals of overlap with the other signal (the one with which the convolution is being calculated) and evaluate the integral of the product of the two signals over these overlapping intervals. In other words, convolution can be interpreted as a measure of the "similarity" or "overlap" of the two signals over their defined intervals.

In the graphs from *Fig. 1*, you can see the shifting of the rectangular signal (in red) along the time axis and its overlap with the parabolic signal (in blue). Initially, the convolution product is zero (*Fig. 1-a*) because the two signals do not overlap at all. As the red signal slides from left to right over the blue signal, the area of overlap gradually increases (the area marked in yellow in *Fig. 1-b, 1-c*), reaching a maximum at some point, followed by a progressive decrease in the overlap area (and hence the convolution integral). Simultaneously, as the overlap occurs, the graph of the convolution function (the black line starting at -0.5) is progressively drawn. The value of this function at any given moment (i.e., the value on the black graph) represents the common overlapping area of the two graphs (red and blue) at that moment — that is, the value of the convolution integral calculated at that point. For a dynamic and complete visualization of the graphs in *Fig. 1*, we recommend visiting the webpage https://en.wikipedia.org/wiki/Convolution.





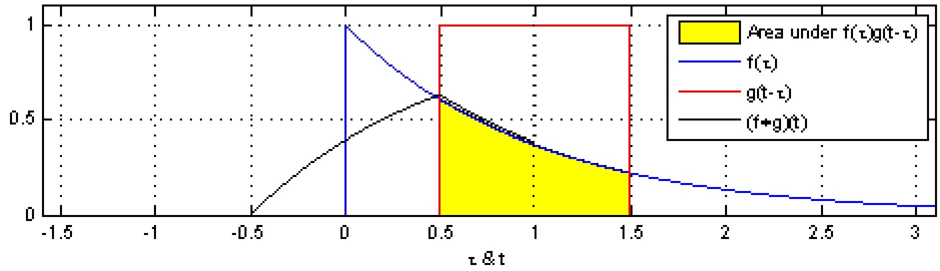


Fig. 1 The convolution of a rectangular signal with an exponential (parabolic) signal: a - top; b - middle; c - bottom

Examples of Convolution for Different Signals

Example 1 – Convolution of a Rectangular Signal with Itself

Let f(t) be a function defined as follows:

$$f(t) = \begin{cases} 1; & 0 \leq t \leq 0.1 \\ 0; & elsewhere \end{cases}$$

To determine the convolution of the above signal with itself, the following Matlab/Octave code can be used:

```
clc;
clear all;
close all;

tstart = 0;
tstop = 0.1;
tpas = 0.0001;

t = tstart: tpas : tstop;
x = ones (1,1001);

subplot(3, 1, 1);
plot(t, x, 'linewidth', 3);
axis ([-0.102 0.212 0 1.2]); grid;

h = ones (1,1001);
subplot(3, 1, 2);
plot(t, h, 'linewidth',3);
axis ([-0.102 0.212 0 1.2]); grid;

t2 = 2*tstart: tpas: 2*tstop;
y = conv(x, h) * tpas;
```

```
subplot (3, 1, 3);
plot(t2, y, 'r', 'linewidth', 2);
axis ([-0.102 0.312 0 0.12]); grid;
```

Code Explanation:

Lines 1 to 3 reset the Matlab/Octave environment, close existing figures, etc.
Lines 5 to 7 initialize the interval bounds where the function is non-zero, as well as the step size to be used for calculating values (the distance between two points in the interval from tstart to tstop).
Lines 9 and 10 define the time value vector (t) and the corresponding values (the x and h vectors, which have the same dimensions as the t vector). See also Fig. 2 below.

Name	Klasse	Dimension	Wert	Attribut
h	double	1x1001	1:0:1	
t	double	1x1001	0:0.0001:0.1	
t2	double	1x2001	0:0.0001:0.2	
tpas	double	1x1	0.00010000	
tstart	double	1x1	0	
tstop	double	1x1	0.10000	
x	double	1x1001	1:0:1	
y	double	1x2001	[0.00010000, 0.0...	

Fig. 2 The Workspace area in the Octave/Matlab environment where the allocated variables are defined.

Lines 12 to 20 plot the two functions *x* and *h* (generated in line 16).
Lines 21 and 22 generate the convolution $y=x*h$; note that the support t2 is non-zero over a different interval than t.
Lines 24 to 26 plot the convolution result (the last plot below, in red).

Observations:

The vectors *h*, *t*, and *x* each contain 1001 elements, as shown in *Fig. 2*. This results from the following combined specifications:

a) tstop−tstart=0.1
b) (tstop−tstart)/tpas=0.1/0.0001=1000
c) The interval is closed at both ends in this example, so an additional value is added to the 1000 existing elements, resulting in 1001 elements in the vector.

The vectors *t2* and *y* contain 2001 elements. The reasoning is similar to the one described earlier. The double number of elements (2000 versus 1000) is justified by considering Property 4 (Duration) of the convolution integral mentioned earlier in this platform.
Fig. 3 shows the graphical representation of the generated signals as well as their convolution (the last plot in the image, in red). The maximum value of the convolution is 0.1, which occurs precisely at the point of maximum overlap of

the two signals (as seen in the top and middle plots of *Fig. 3*). The common area in this case is 0.1×1=0.1. The "moment in time" when this happens is 0.1. Practically, one of the two signals considered is "rotated" and "moved" over the other from left to right on the graph (see also the theoretical introduction above). When the "shifted" graph exceeds the maximum overlap point, the convolution value starts to decrease (the descending ramp of the convolution triangle in *Fig. 3*). When the two signals no longer overlap at all (i.e., t2>0.2), the convolution becomes 0.

Exercise: Modify the code so that one of the signals *x* or *h* has a duration of 0.05 (i.e., $0 \le t \le 0.05$). The other signal remains defined as before. Display the graph of the convolution function in this new situation. Comment on the result obtained. Also, adjust the graphical representations so that they align at zero vertically.

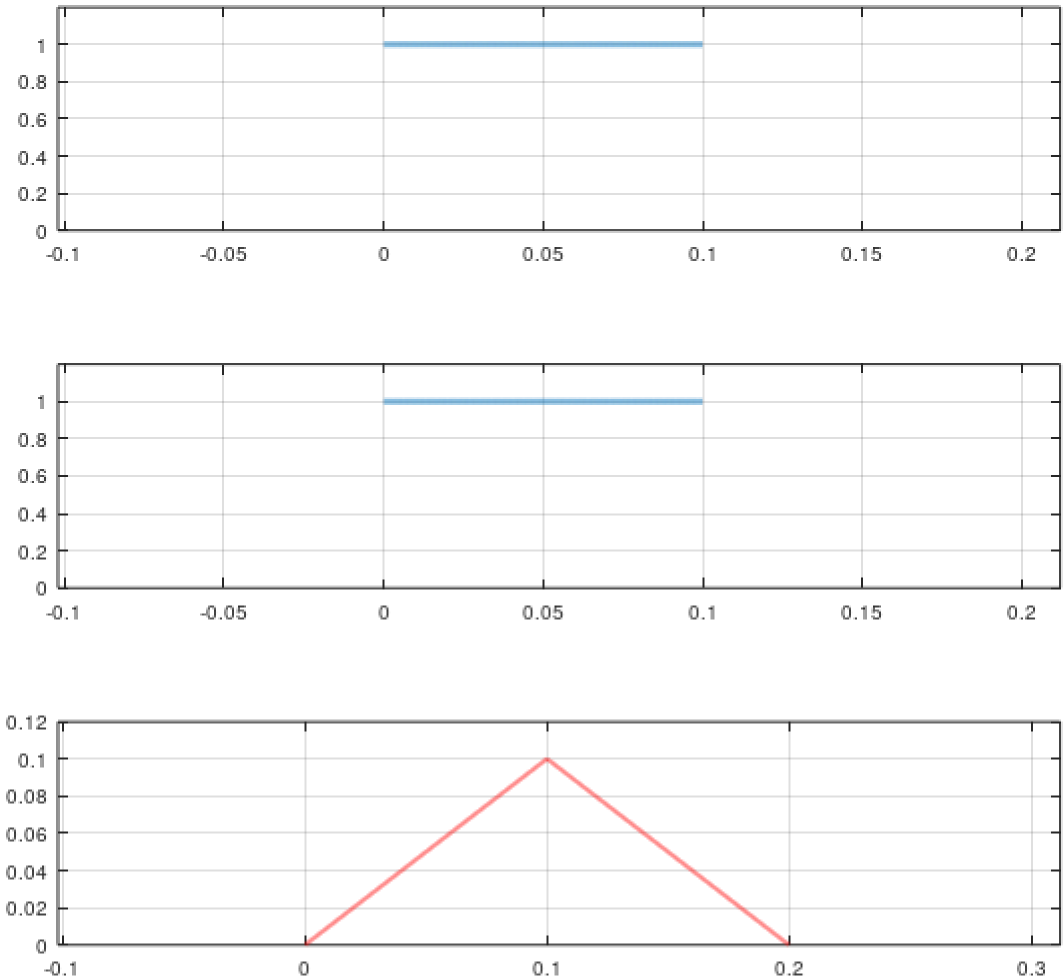


Fig. 3 Convolution of a Rectangular Signal with Itself – (Top and Middle: x and h); Bottom – Convolution y=x*h

Example 2 – Decreasing Ramp Signal with Decreasing Exponential Signal
Let x(t)x(t)x(t) and h(t)h(t)h(t) be two functions defined as follows:

$$x(t) = \begin{cases} 1 - 10t; & 0 \le t \le 0.1 \\ 0; & elsewhere \end{cases}$$

$$h(t) = \begin{cases} e^{-ft}; & 0 \le t \le 0.1 \\ 0; & elsewhere \end{cases}$$

To determine the convolution of the above signals x(t)x(t)x(t) and h(t)h(t)h(t), you can use the following Matlab/Octave code:

```
clc;
clear all;
close all;

tstart = 0;
tstop = 0.1;
tpas = 0.0001;
f = 100;

t = tstart : tpas : tstop;
x= 1-10*t;

subplot(3, 1, 1);
plot(t, x, 'linewidth', 2);
axis([0 0.1001 0 1]); grid;

h = 1 * exp(-f*t);
subplot(3, 1, 2);
plot(t, h, 'linewidth', 2);
axis([0 0.1001 0 1]); grid;

t2 = 2*tstart: tpas : 2*tstop;
y = conv(x, h) * tpas;

subplot(3, 1, 3);
plot(t2, y, 'r', 'linewidth',2);
axis(); grid;
```

In this example, we have two distinct functions/signals for which the convolution is calculated. The Matlab/Octave function that performs convolution is called conv, and its usage is similar to that shown in the first example. Note the vertical alignment of the axes at zero.

Question: Why is the value at line 23 multiplied by tpas?

In *Fig. 4*, the convolution of the defined signals is graphically represented. The graphical "theoretical" calculation method is similar to that described in the previous example. The temporal support of the convolution aligns with Property 4 (Duration) of the convolution integral.

Exercise: Modify the code so that one of the signals is an increasing ramp instead of a decreasing ramp (i.e., it has a value of 0 at t=0t = 0t=0 and 1 at t=0.1t = 0.1t=0.1). Display the graph of the convolution function in this new situation. Comment on the result obtained. What should be modified in the code to obtain a ”smoother” exponential variation?

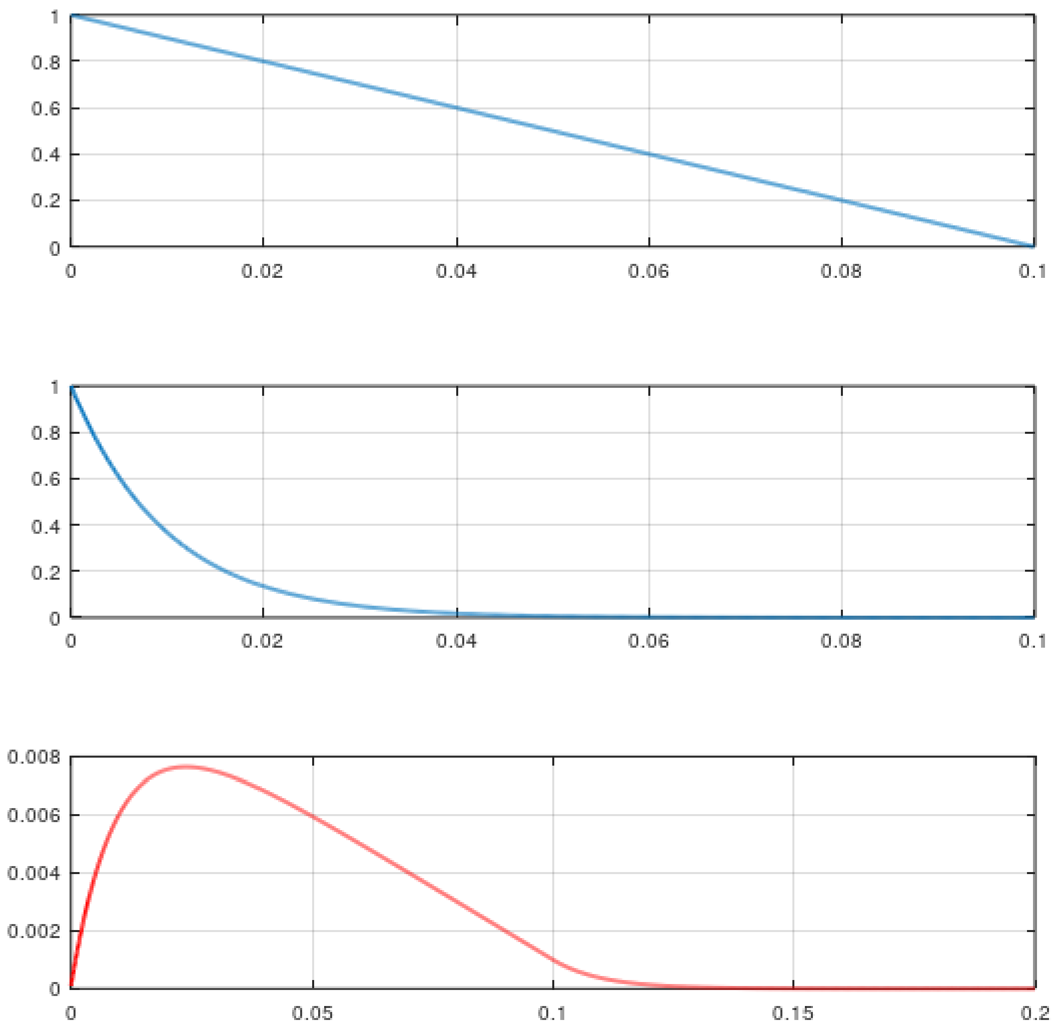


Fig. 4 Convolution of the Ramp Signal with the Exponential Signal – (Top and Middle: x and h); down – convolution y=x*h

The continuous-time correlation function of two signals $x_1(t)$ and $x_2(t)$ is defined by the following relation:

$$x(\tau) = (x_1 * x_2)(\tau) = \int_{-\infty}^{\infty} x_1^*(t)\, x_2(t+\tau)dt \quad ; \quad -\infty < t < \infty$$

In the previous relation, $x_1^*(t)$ represents the complex conjugate of $x_1(t)$, and τ is the "shift" or "delay." The interpretation is as follows: a characteristic of the signal present in x_1 at time t is also found in x_2 at time $t + \tau$.

The correlation function represents a measure of similarity (overlap) between elements of a vector x and shifted (delayed) values of another vector y as a function of the shift. Correlation is often used to search for a specific shorter sequence (subvector with certain values) within a longer signal (vector).

Example – Correlation of an Exponential Signal with Itself (Autocorrelation) – Discrete Time

In Matlab/Octave, the *xcorr* function can be used to determine the correlation (referred to as cross-correlation in English) between two signals. If the two sequences are identical (i.e., the same signal), it is referred to as the autocorrelation of the signal.
The code below determines the autocorrelation of the signal$e^{-n/4}$ using the *xcorr* function.

```
clc;
clear all;
close all;

n = 0:10;
x = exp(-n/4);
y= exp(-n/4);
[z, intarziere] = xcorr(x, y);

subplot(3, 1, 1);
stem (n, x);
axis(); grid; title('x');

subplot(3, 1, 2);
stem (n, y);
axis(); grid; title('y');

subplot(3, 1, 3);
stem (intarziere, z);
axis(); grid; title('z = x*y');
```

The code generates the following graphical sequences for x, y, and their correlation function, i.e., z=x*y. It is observed that in this example, x = y (autocorrelation). For graphical representation in discrete time, the Matlab/Octave function *stem* is used.

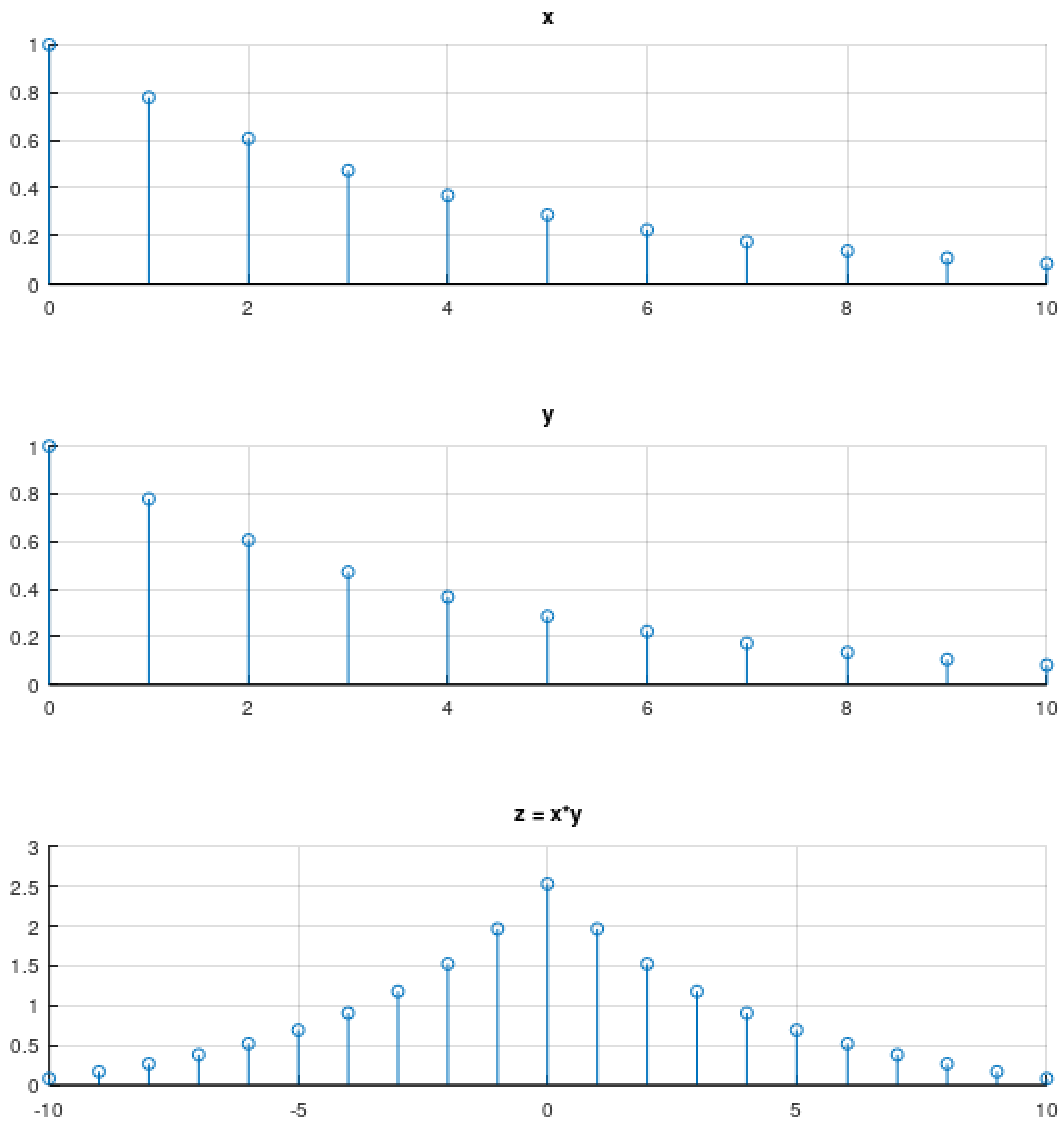


Fig. 5 Autocorrelation of the Exponential Signal

Observations:

a) The maximum of the correlation function is achieved at value 0, i.e., when the two signals are not shifted relative to each other.

b) As one signal is shifted relative to the other, their correlation decreases. Additionally, the symmetry of the correlation values around the y-axis is observed (the evenness of the correlation function).

c) For displacement values greater than 10, the correlation becomes 0 because the two signals no longer overlap at all.

Homework:

1. Define: $x(t) = t; t \in [0,1]$ și $y(t) = t^2; t \in [0,1]$.
2. Define: $x(t) = t; t \in [0,1]; x(t) = t^2; t \in (1,2]$ and $y(t) = t^{-3}; t \in [1,2]$.

3. Define: $x(t) = t; t \in [1,2];$ and $y(t) = t^2; t \in [0,2]$
4. Define the functions: $x(t) = a * t + b; t \in [1,2]$ such that $x(1) = 2\ ;\ x(2) = 1$ and $y(t) = t^2; t \in [0,2]$. Determine $x(t)$.
5. Define the functions: $x(t) = a * t + b; t \in [0,1]$ such that $x(0) = 0\ ;\ x(1) = 4$ și $y(t) = c * t + d; t \in [1,2]$ such that $x(1) = 3\ ;\ x(2) = 0$. Determine $x(t)$.
6. Define the functions: $x(t) = a * t + b; t \in [0,2]$ such that $x(0) = 3\ ;\ x(2) = 1$ and $y(t) = c * t + d; t \in [1,2]$ such that $x(1) = 3\ ;\ x(2) = 0$
7. Define: $x(t) = 2^{-t}; t \in [1,2];$ and $y(t) = 3^{-t}; t \in [1,2]$.

a) For the signals defined above, implement the code necessary to generate the convolution function and plot the three signals (x, y, and their convolution). Where necessary, first determine the parameters a, b, c, and d of the corresponding signals.

b) Redefine the above signals in discrete numerical intervals similar to the previous example (adjusting the intervals of definition of the functions with new values if necessary). Implement the correlation function between them. The step size between consecutive values on the discrete time axis should be chosen suitably for representation with the stem function.

Bibliography

1. Mateescu, Adelaida, Dumitriu, N., Stanciu, L., **Semnale, circuite și sisteme**, Teora, București, 2001.