Classes 8-9 (4 hours). Visualization and manipulation in Matlab.

1. Creating animations

In the Matlab environment, we can create different types of graphs using equations for functions. If the function has additional parameters it is possible to manipulate such a graph and visualize its form after changing the parameters in the graphical window.

Visualization and manipulation can take place in several ways. Below it is describes the method for creating a matrix frames (`getframe)` and the method that uses the command `comet.`

2. Create a frame matrix for graphs

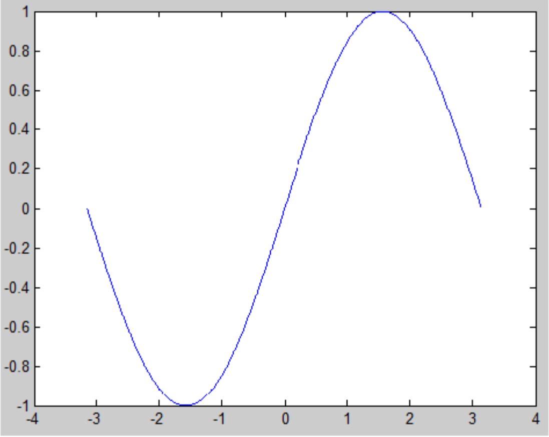
Since each drawn graph is in a figure element, we can use the `getframe` command to write, for example, the value of the entire window to the matrix. In this way we can create a sequence of animated graphs of the function plot in the loop for the variation of some of its parameters. The concept seems simple and intuitive. Let's look at the loop that can be created frame matrix:

```
for k = 1:10
drawing a graph (); // Sample drawing function - fence image to matrix.
M(k) = getframe;
end
```

Then the following command will reproduce the resulting movie:

```
 movie(M,20)
```

Consider the function sin(x), where x varies from $[-\pi;\pi]$. Plot function is below



If we multiply the sine function by a scalar value, we will change its amplitude to smaller or larger.

Example 1. Create an animation of function graph for k sin(x), where

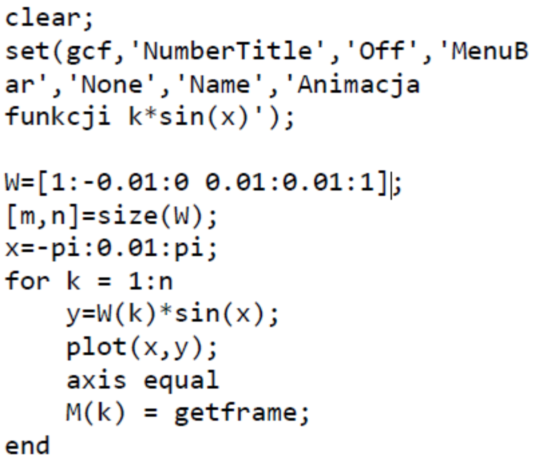
`k = 1 0.99 0.98 ... 0.99 1`.

First, create a frame array using the `getframe` command, and then use the command:

```
 movie(M,n,fps);
```

pass frame matrix **M**, the number of times for the animation **n** and specify the number of frames per second fps.

Below is a script that creates an **M** matrix:



```
clear;
set(gcf,'NumberTitle','Off','MenuB
ar','None','Name','Animacja
funkcji k*sin(x)');

W=[1:-0.01:0 0.01:0.01:1];
[m,n]=size(W);
x=-pi:0.01:pi;
for k = 1:n
    y=W(k)*sin(x);
    plot(x,y);
    axis equal
    M(k) = getframe;
end
```

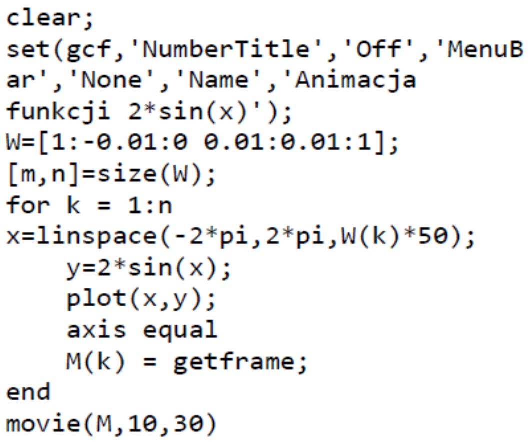
In addition, the `set()` function modifies our window with a graph. The `clear` command clears unnecessary variables in the environment. Now, matrix **M** is ready so just in the console type:

```
movie(M,10,120)
```

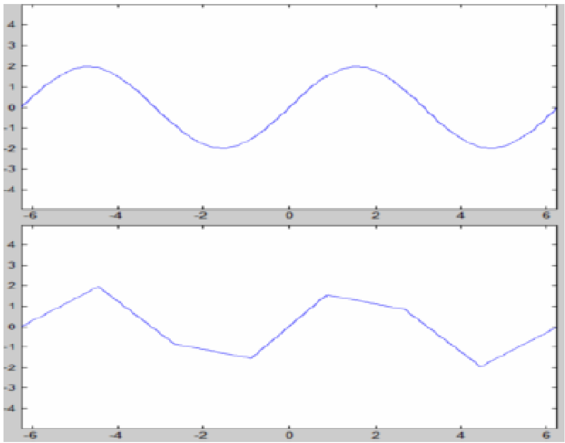
to see the animation. The animation speed can be manipulated by decreasing the number of elements in the vector **k** or by manipulating the **fps** parameter.

Example 2. Using the animation we can also look at the effect of the precision of drawing a graph (number of `par(x,y)`) on its appearance.

Consider the following script:



```
clear;
set(gcf,'NumberTitle','Off','MenuB
ar','None','Name','Animacja
funkcji 2*sin(x)');
W=[1:-0.01:0 0.01:0.01:1];
[m,n]=size(W);
for k = 1:n
x=linspace(-2*pi,2*pi,W(k)*50);
    y=2*sin(x);
    plot(x,y);
    axis equal
    M(k) = getframe;
end
movie(M,10,30)
```

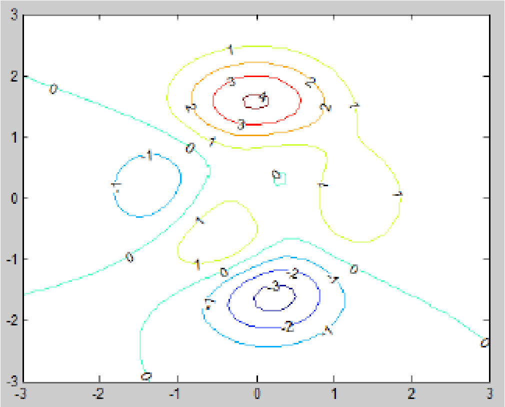


Placing the variation parameter as the third argument of `linspace(a,b,n)` allows you to observe the accuracy of the graph, where `a` and `b` define the range and `n` the number of elements in the resulting vector.

Example 3. Create a contour plot for any surface function.

Use the ready-to-use function `contour()` forming a contour plot. As an example of the surface has been selected ready graph peaks.

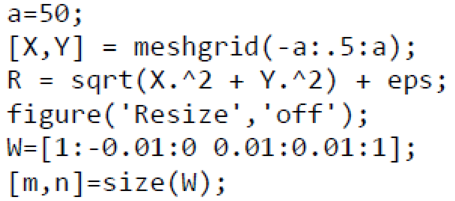
```
W=[1:-0.01:0 0.01:0.01:1];
[m,n]=size(W);
for k = 1:n
x = linspace(-2*pi,2*pi);
y = linspace(0,4*pi);
[X,Y,Z] = peaks;
Z=Z*W(k);
% figure
contour(X,Y,Z,'ShowText','on')

    M(k) = getframe;
end
movie(M,10,30)
```



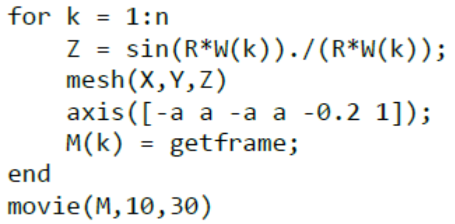
During the percentage change of the matrix **Z** we can observe what happens to the surface on the contour plot.

Example 4. Show grid animation of any surface plot when changing the selected parameter.

For the simulation the `sinc` function was selected.



```
a=50;
[X,Y] = meshgrid(-a:.5:a);
R = sqrt(X.^2 + Y.^2) + eps;
figure('Resize','off');
W=[1:-0.01:0 0.01:0.01:1];
[m,n]=size(W);
```



```
for k = 1:n
    Z = sin(R*W(k))./(R*W(k));
    mesh(X,Y,Z)
    axis([-a a -a a -0.2 1]);
    M(k) = getframe;
end
movie(M,10,30)
```

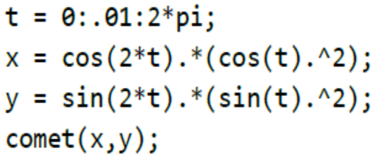
This example manipulates the values of the parameter **R**. The values, as in the preceding examples, are reduced from 100% to 0% and back up to 100%. It is important to note that during the graph animation the axes can be modified. If you do not want the axes to change their scale, use the `axis` command to set the axes constant.

This allows us to observe what happens to the graph when manipulating the selected data.

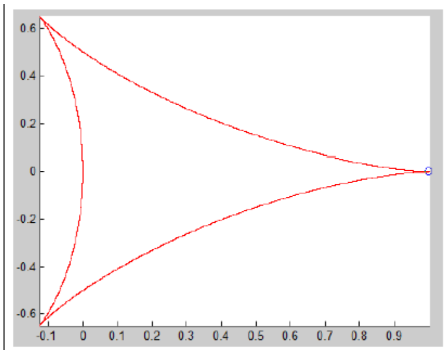
3. Create graph animations

You can also use the `comet` command to create graph animations.

Example 5. Animation of the motion of the point after the determined parametric curve.



```
t = 0:.01:2*pi;
x = cos(2*t).*(cos(t).^2);
y = sin(2*t).*(sin(t).^2);
comet(x,y);
```



This example generates an animation of a point moving in accordance with a parametric equations.

$$\begin{cases} x = \cos(2t)\ \cos^2(t), \\ y = \sin(2t)\ \sin^2(t), \end{cases}$$

where $t \in [0, 2\pi]$.

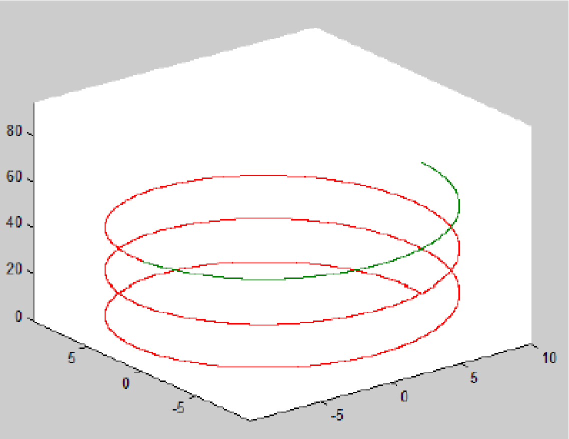
Matlab also implements a 3D variant of this command: `comet3`.

Example 6. Show point movement after circular helix given by equation:

$$\begin{cases} x = a * \cos(t)\,, \\ y = a * \sin(t)\,, \\ \quad z = b * t \end{cases}$$

for $a = 10, b = 4$ and $t \in [0, 2\pi]$.

```
t=0:0.001:10*pi;
a=10;
b=4;
x = a*cos(t);
y = a*sin(t);
z=b*t;
comet3(x,y,z);
```



Exercises

Exercise 1. Create a variance animation of the parameter k for:

- the function $f(x) = x^k$ for $k \in [2,4, \dots ,20]$ and for $k \in [1,3, \dots ,19]$;
- trigonometric function $f(x) = \sin\,(x + k * 2 * pi)$ for $k \in [0,1]$.

Exercise 2. Using example 4, where the grid is presented on the graph, create an animation for the surface of the graph. Use the `surf()` command.

Exercise 3. Show of the manipulation of the parameter k for the surface function

$$z(x, y) = \cos(k\,x) + \cos\,(k\,y)$$

on the interval $(x, y) \in [-20, 20]$ and for $k \in [1\ \ 0.99\ \ 0.98\ \ \ldots 0.99\ \ 1]$. Use the `figure()` command to display the animation on the maximized window:

```
figure('Resize','off','units','normalized','outerposition',[0 0 1 1])
```

Exercise 4. Show the point movement on the ellipse with center $(0,0)$ and semimajor axis $a = 2$, semiminor axis $b = 6$.

Exercise 5. Show the point motion on the curve that defined in the parametric form

$$\begin{cases} x = \cos(\,t) + \cos\dfrac{7 * t}{2} + \sin\dfrac{14 * t}{2}, \\ y = \sin(\,t) + \sin\dfrac{7 * t}{2} + \cos\dfrac{14 * t}{2}, \end{cases}$$

where $t \in [0, 2\pi]$. Hint: the smaller the jump for *t*, the slower it will be to move the point.

Exercise 6. It is given the elongated cycloid by the formulas

$$x(t) = a * t - \mathrm{b} * \sin(t),\ y(t) = a - b * \cos(t)$$

for $a = 4$ and $b = 9$. Show the point motion on the given curve for $t \in [0, 20\pi]$.

Exercise 7. Show a point moving through the archimedean spiral parameter:

$$\begin{cases} x = a * t * \cos(\,t)\,, \\ y = a * t * \sin(\,t) \end{cases}$$

for $a = 0.1$ and $t \in [0, 50]$.

Exercise 8. Show point motion on Lissajous curve:

$$\begin{cases} x = A * \cos(\mathrm{a} * t + b)\,, \\ y = B * \cos(\mathrm{c} * t + d) \end{cases}$$

for $A = 20, B = 10$ and $\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 10 & 3 \\ 6 & 4 \end{bmatrix}$.

Exercise 9. The Viviani curve is given (spherical lemniscus):

$$\begin{cases} x = a * \cos^2(\,t)\,, \\ y = a * \sin(\,t) * \cos\,(t), \\ z = a * \sin\,(t) \end{cases}$$

for $a = 5$. Create a simulation of the point movement through the above curve.