

Politechnika Gdańska
Wydział Elektrotechniki i Automatyki
Katedra Inżynierii Systemów Sterowania

Teoria sterowania

Grafika w środowisku MATLAB

Materiały pomocnicze do ćwiczeń laboratoryjnych 1 – Część 4

Opracowanie:

Michał Grochowski, dr inż.

Robert Piotrowski, dr inż.

Łukasz Michalczyk, mgr inż.

Gdańsk

Grafika dwuwymiarowa – podstawy

W MATLAB'ie istnieje kilka grup funkcji graficznych:

- funkcje przeznaczone do prezentacji danych w postaci wykresów dwu i trójwymiarowych,
- funkcje związane z usuwaniem rysunku, zmianą skali, dodawaniem napisów itp.,
- funkcje umożliwiające rysowanie linii, wielokątów itp.,
- funkcje niskiego poziomu pozwalające na dowolne kształtowanie wyglądu tworzonego rysunku.

Podstawowym poleceniem służącym do graficznej prezentacji danych matematycznych jest polecenie **plot**, które powoduje narysowanie wykresu funkcji jednej zmiennej. Przykładowo, narysowanie funkcji $y=x$ dla $x \in \langle 0,10 \rangle$ można uzyskać po wprowadzeniu następującego programu:

```
x=0:0.1:10;  
y=x;  
plot(x,y)
```

Inną formą zapisu funkcji **plot**, umożliwiającą wykreślenie dwóch funkcji lub więcej na jednym wykresie jest:

```
plot (x1, y1, x2, y2, ...)
```

Funkcja **plot** może być użyta również z innymi parametrami (określającymi np. kolor wykresu – tabela 1, rodzaj użytej linii – tabela 2 czy oznaczenia punktów – tabela 3) szczegółowo opisanymi w pomocy MATLAB'a. Przykładowo, narysowanie funkcji $y=x$ dla $x \in \langle 0,10 \rangle$ o linii koloru czerwonego i rodzaju kreskowo – punktowego można uzyskać następująco:

```
x=0:0.1:10;  
y=x;  
plot(x,y, 'r-.')
```

Tabela 1. Kolory linii wykresu

Symbol	Kolor
b	blue (niebieski)
g	green (zielony)
r	red (czerwony)
c	cyan (błękitny)
m	magenta (purpurowy)
y	yellow (żółty)
k	black (czarny)
w	white (biały)

Tabela 2. Rodzaje linii wykresu

Symbol	Opis
–	ciągła
:	punktowa
–.	kreskowo – punktowa
--	kreskowa

Tabela 3. Oznaczenia punktów wykresu

Symbol	Opis
.	punkt •
o	okrąg o
x	znak x
+	znak +
*	gwiazdka
s	kwadrat
d	znak ?
v	trójkąt ∇
?	trójkąt ?
>	trójkąt prawy
<	trójkąt lewy
p	gwiazda pięciokątna
h	gwiazda sześciokątna

Okno graficzne można oczyścić po wywołaniu funkcji **clf**, natomiast funkcja **close** zamyka okno.

Inne wybrane funkcje związane z grafiką dwuwymiarową przedstawia tabela 4.

Tabela 4. Wybrane funkcje związane z grafiką dwuwymiarową

Funkcje	Opis
subplot(m,n,p)	Umieszczenie kilku wykresów obok siebie w jednym oknie (m – liczba wykresów w pionie, n – liczba wykresów w poziomie, p – numer wykresu, który zostanie narysowany najbliższym wywołaniem funkcji plot)
axis([x _{min} x _{max} y _{min} y _{max}])	Skalowanie wykresu (x _{min} x _{max} y _{min} y _{max} – zakresy skal na poszczególnych osiach układu współrzędnych)
hold on (lub hold off)	Hold on – pozwala rysować nowy wykres na tle starego. Hold off – przed narysowaniem nowego wykresu stary wykres jest wymazywany. Polecenie hold bez parametrów na zmianę włącza i wyłącza czyszczenie ekranu
loglog(x,y)	Rysowanie wykresu z użyciem skal logarytmicznych na obu osiach
semilogx(x,y)	Rysowanie wykresu z użyciem skali logarytmicznej na osi odciętych
semilogy(x,y)	Rysowanie wykresu z użyciem skali logarytmicznej na osi rzędnych

Należy zaznaczyć, że praktycznie każda z przedstawianych funkcji, umożliwia jej stosowanie z dodatkowymi parametrami szczegółowo opisanymi w pomocy MATLAB'a. Do szybkiego rysowania wykresów zależności funkcyjnych służy funkcja **fplot**.

Każdy wykres można opisać. Służą do tego między innymi następujące funkcje:

- **xlabel('tekst')** – wypisuje łańcuch znaków *tekst* pod osią x
- **ylabel('tekst')** – wypisuje łańcuch znaków *tekst* pod osią y
- **text(x,y,'tekst')** – wypisuje łańcuch znaków *tekst* w miejscu określonym przez współrzędne x i y
- **gtext('tekst')** – wypisuje łańcuch znaków *tekst* w miejscu wskazanym myszką

- **title('tekst')** – wypisuje tytuł wykresu w postaci łańcucha znaków *tekst* i umieszcza go nad wykresem
- **legend('tekst1','tekst2'. ... , 'tekstn')** – wypisuje legendę do wykresu w postaci łańcucha znaków *tekst1, tekst2, ... tekstn*
- **grid on (lub grid off)** – narysowanie pomocniczej siatki współrzędnych (grid on) lub jej wyłączenie (grid off). Polecenie grid bez parametrów powoduje przełączanie wyświetlania siatki

Przykład 1

Rozważmy następujące zadanie:

Napisz m-plik kreślący na jednym wykresie następujące funkcje: $y_1 = k_1 \cdot x$, $y_2 = x^{k_2}$ dla $x \in <0,10>$. Przyjmij następujące wartości parametrów: $k_1 = 3$, $k_2 = 2$ i załóż możliwość zmiany ich wartości w m-pliku. Podpisz osie, umieść siatkę, tytuł wykresu i legendę.

Przykładowe rozwiązanie:

```
% Czyszczenie
clear;
clc;

% Wyświetlenie tekstu na ekranie
disp('M plik kreslacy wykresy dwoch funkcji: y1 i y2')
disp('')

% Okreslenie wartosci x
x=-10:0.1:10;

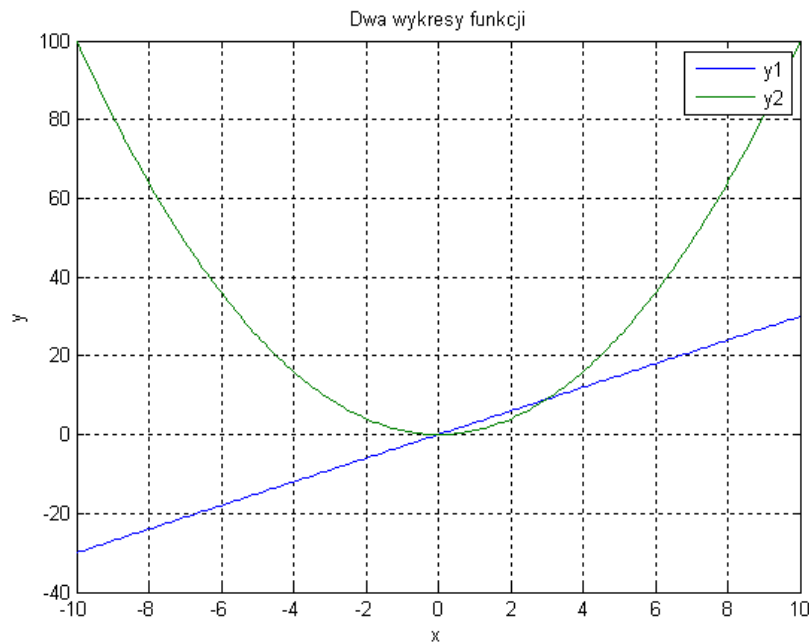
% Wprowadzenie danych
k1=3;
k2=2;

%Obliczenie wartosci pierwszej funkcji
y1=k1*x;

%Obliczenie wartosci drugiej funkcji
y2=x.^k2;

%Wykreslenie dwoch funkcji y1 i y2 na jednym wykresie, narysowanie siatki,
podpisanie wykresu i obu osi
plot(x,y1,x,y2);
grid;
title('Dwa wykresy funkcji');
xlabel('x');
ylabel('y');
legend('y1','y2')
```

Wynik działania m-pliku:



Rys. 1. Wynik działania m-pliku z Przykładu 1

Przykład 2

Napisz m-plik kreślący na dwóch osobnych wykresach w jednym oknie następujące funkcje: $y_1 = k_1 \cdot \sin(k_1 \cdot x)$, $y_2 = \cos(k_2 \cdot x)$ dla $x \in \langle -2\pi, 2\pi \rangle$. Przyjmij następujące wartości parametrów: $k_1 = 0.5$, $k_2 = 1$. Podpisz osie, wyskaluj wykres, umieść siatkę i tytuł wykresu.

```
% Czyszczenie
clear;
clc;

% Wyświetlenie tekstu na ekranie
disp('M plik kreslacy wykres dwoch funkcji: y1 i y2 obok siebie w jednym oknie')
disp('')

% Okreslenie wartosci x
x=-2*pi:0.1:2*pi;

% Wprowadzenie danych
k1=0.5;
k2=1;

%Obliczenie wartosci pierwszej funkcji
y1= k1*sin(k1*x);

%Obliczenie wartosci drugiej funkcji
y2=cos(k2*x);

%Wykreslenie dwoch funkcji y1 i y2 obok siebie w jednym oknie, skalowanie wykresu, narysowanie siatki, podpisanie wykresu i obu osi
subplot(1,2,1)
plot(x,y1);
```

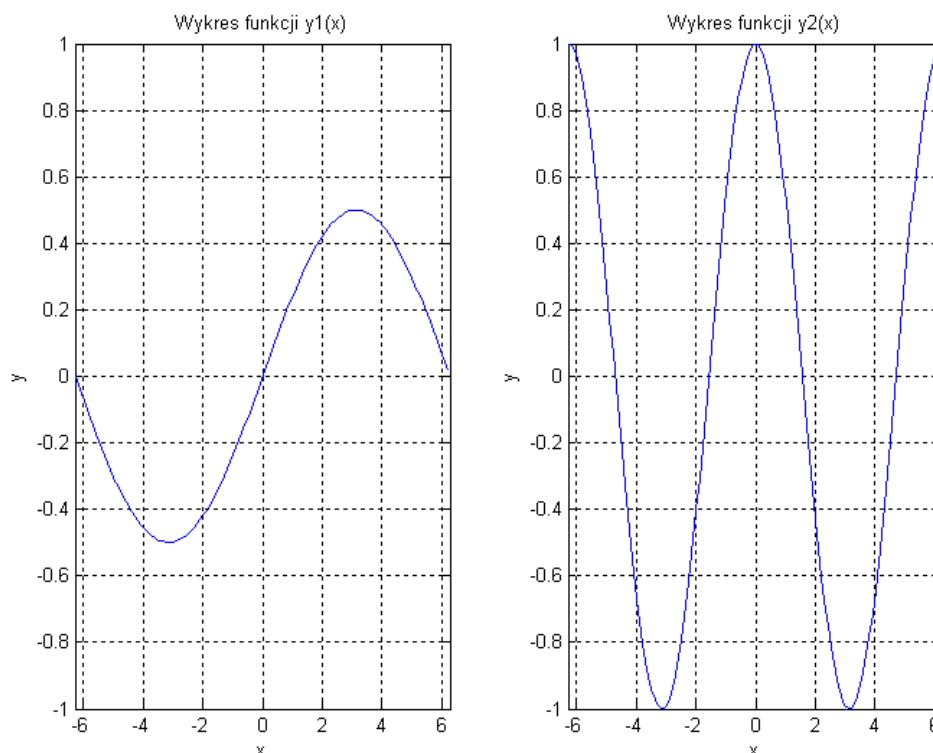
```

axis([-2*pi 2*pi -1 1]);
grid;
title('Wykres funkcji y1(x)');
xlabel('x');
ylabel('y');

subplot(1,2,2)
plot(x,y2);
axis([-2*pi 2*pi -1 1]);
grid;
title('Wykres funkcji y2(x)');
xlabel('x');
ylabel('y');

```

Wynik działania m-pliku:



Rys. 2. Wynik działania m-pliku z Przykładu 2

Grafika dwuwymiarowa – rozwinięcie

W MATLAB'ie możliwe jest również rysowanie wykresów w innych układach współrzędnych:

Wykres w układzie biegunowym

Punkt (x,y) w układzie kartezjańskim reprezentowany jest przez punkt (θ,r) w układzie biegunowym, gdzie:

$$\begin{aligned}
 x &= r \cdot \cos(\theta) \\
 y &= r \cdot \sin(\theta)
 \end{aligned}$$

gdzie $\theta \in \langle 0, 2\pi \rangle$.

Do rysowania wykresów w układzie biegunowym służy funkcja **polar(theta,ro)**

Przykład 3

Napisz m-plik kreślący wykres funkcji: $y_1 = \cos(k_1 \cdot x)$ dla $x \in \langle 0, 2\pi \rangle$. Przyjmij następującą wartość parametru: $k_1 = 2$. Umieść siatkę i tytuł wykresu.

```
% Czyszczenie
clear;
clc;

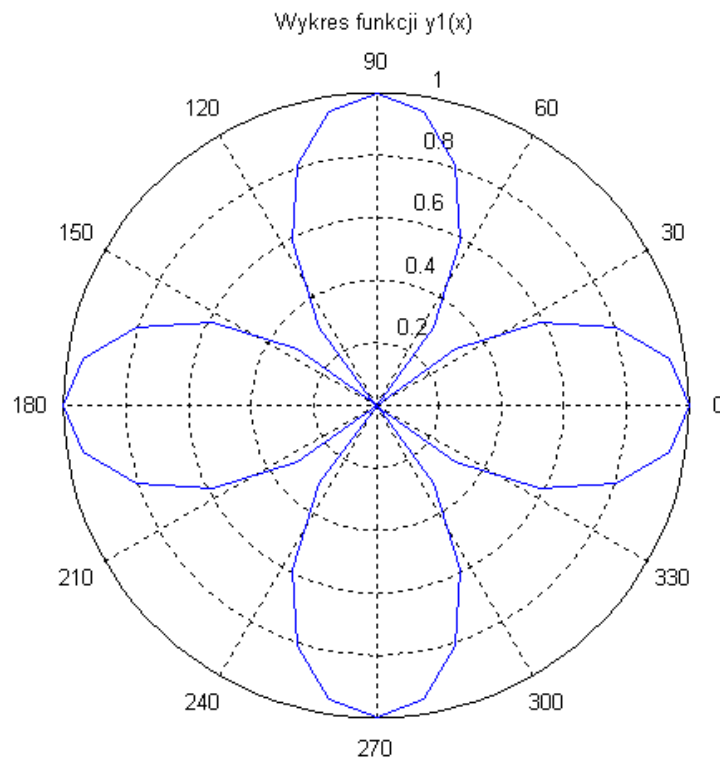
% Wyświetlenie tekstu na ekranie
disp('M plik kreslacy wykres w ukkladzie biegunowym')
disp('')

% Okreslenie wartosci x
x=0:0.05*pi:2*pi;

%Obliczenie wartosci  funkcji
y= cos(2*x);

%Wykreslenie funkcji y1(x) , narysowanie siatki, podpisanie wykresu
polar(x,y1);
grid;
title('Wykres funkcji y1(x)');
```

Wynik działania m-pliku:



Rys. 3. Wynik działania m-pliku z Przykładu 3

Wykres słupkowy

Jest to jedna z graficznych metod służąca przedstawianiu i prezentacji danych statystycznych za pomocą prostokątów w dwuwymiarowym układzie kartezjańskim.

Do rysowania wykresów słupkowych służy funkcja **bar(x,y)**

Przykład 4

Napisz m-plik kreślący wykres słupkowy funkcji $y(x)$ gdzie $x=\{1,2,3,\dots,10\}$ dla wygenerowanych 10 liczb pseudolosowych o rozkładzie jednostajnym na przedziale $y \in <0,1 >$. Umieść tytuł wykresu i podpisz osie.

```
% Czyszczenie
clear;
clc;

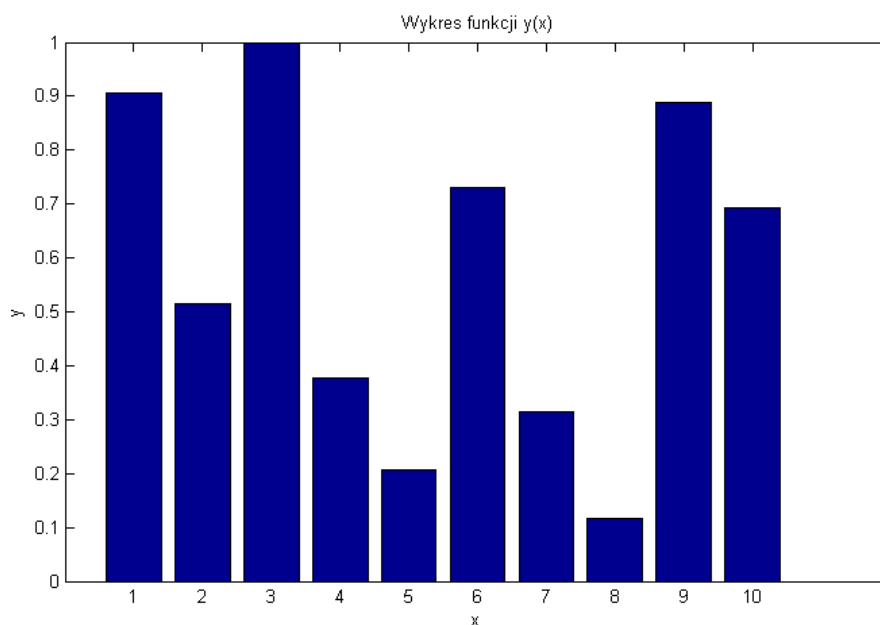
% Wyszwietlenie tekstu na ekranie
disp('M plik kreslacy wykres slupkowy')
disp('')

% Okreslenie wartosci x
x=1:1:10;

% Generowanie wektora 10 liczb pseudolosowych o rozkladzie jednostajnym
y= rand(1,10);

%Wykreslenie funkcji y(x), podpisanie wykresu i osi
bar(x,y);
title('Wykres funkcji y(x)');
xlabel('x');
ylabel('y')
```

Wynik działania m-pliku:



Rys. 4. Wynik działania m-pliku z Przykładu 4

Histogram słupkowy

Jest to metoda służąca przedstawieniu częstości występowania określonej wielkości za pomocą szeregu prostokątów w dwuwymiarowym układzie kartezjańskim.

Do rysowania histogramów słupkowych służy funkcja **hist(y,x)**

Przykład 5

Napisz m-plik kreślący histogram słupkowy funkcji $y(x)$ gdzie $x=\{0,1,2,3,\dots,100\}$ dla wygenerowanych 1000 liczb pseudolosowych o rozkładzie normalnym o wartości średniej 50 i wariancji 10. Umieść tytuł wykresu, wyskaluj oś x i podpisz osie.

```
% Czyszczenie
clear;
clc;

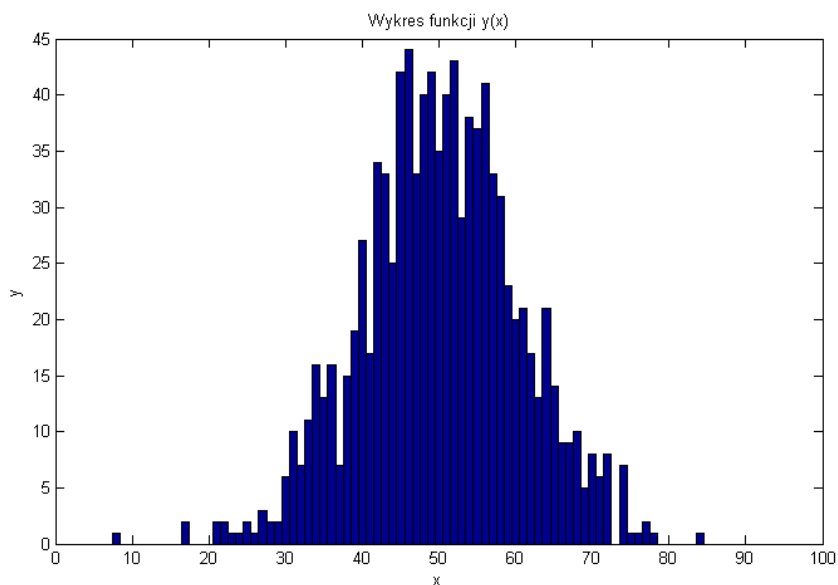
% Wświetlenie tekstu na ekranie
disp('M plik kreslacy histogram slupkowy')
disp('')

% Okreslenie wartosci x
x=0:100;

% Generowanie wektora 1000 liczb pseudolosowych o rozkładzie normalnym
y= 50+10*randn(1,1000);

%Wykreslenie funkcji y(x), podpisanie wykresu i osi, wyskalowanie osi x
hist(y,x);
title('Wykres funkcji y(x)');
xlabel('x');
ylabel('y');
xlim([0 100]);
```

Wynik działania m-pliku:



Rys. 5. Wynik działania m-pliku z Przykładu 5

Drugim przykładem histogramu jest histogram kołowy, do rysowania którego służy funkcja **rose(t,x)**

Wykres schodkowy

Do rysowania wykresu schodkowego służy funkcja **stairs(x,y)**.

Przykład 6

Napisz m-plik kreślący wykres schodkowy funkcji $y = \sin(x)$ dla $x \in (0, 3\pi)$ z krokiem 0.25π .

```
% Czyszczenie
clear;
clc;

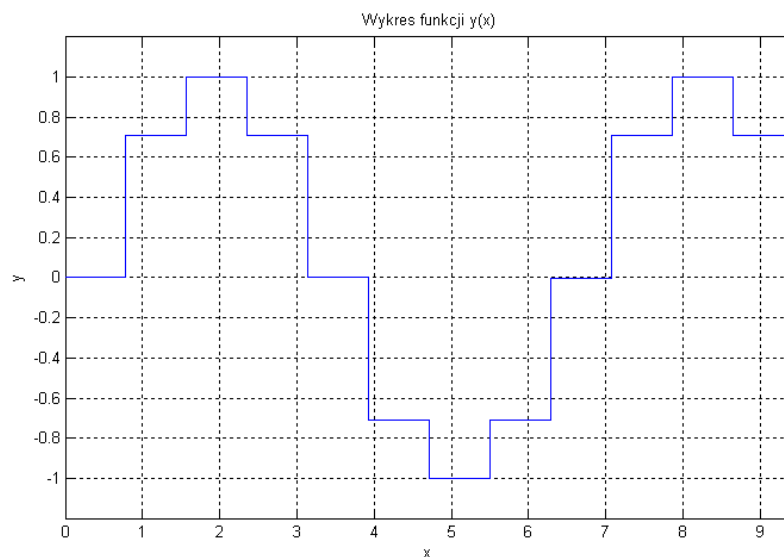
% Wyświetlenie tekstu na ekranie
disp('M plik kreslacy wykres schodkowy')
disp('')

% Okreslenie wartosci x
x=0:0.25*pi:3*pi;

% Obliczenie wartości funkcji
y=sin(x);

%Wykreslenie funkcji y(x), podpisanie wykresu i osi, wyskalowanie osi,
utworzenie siatki
stairs(x,y);
title('Wykres funkcji y(x)');
xlabel('x');
ylabel('y');
axis([0 3*pi -1.2 1.2])
grid;
```

Wynik działania m-pliku:



Rys. 6. Wynik działania m-pliku z Przykładu 6

Wykresy w dziedzinie liczb zespolonych

Dane, które zawierają wartości zespolone można rysować przy użyciu specjalnych funkcji matematycznych wyodrębniających ich części rzeczywiste i urojone. Przykładowymi użytecznymi funkcjami są: **plot(real(z), imag(z))** wykreślająca wykres w przestrzeni zespolonej, gdzie **real(z)** to część rzeczywista liczby zespolonej **z**, **imag(z)** to część urojona liczby zespolonej **z**. Polecenia **compass(x,y)**, **feather(x,y)** wykreślają wykres w postaci wektorów w kształcie strzałek i mogą być użyte zarówno w dziedzinie liczb zespolonych jak i rzeczywistych.

Przykład 7

Napisz m-plik kreślący wykres w dziedzinie liczb zespolonych $z = x + iy$ dla $x \in <-10,10 >$ i $y = 0.5 \cdot x$.

```
% Czyszczenie
clear;
clc;

% Wyświetlenie tekstu na ekranie
disp('M plik kreslacy wykres w dziedzinie liczb zespolonych')
disp('')

% Okreslenie wartosci x
x=-10:0.1:10;

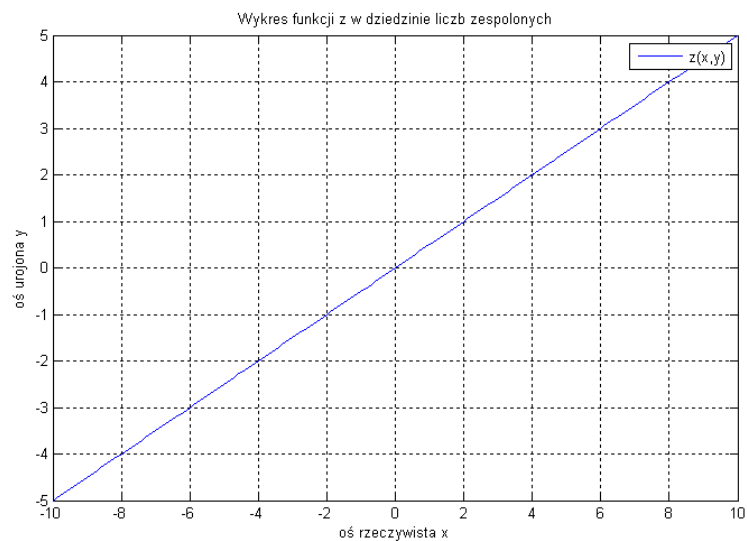
% Wyznaczenie wartosci funkcji y
y=0.5*x

%Obliczenie wartosci funkcji z w dziedzinie zespolonej
z=x+i*y;

%Wykreslenie funkcji z w dziedzinie zespolonej, narysowanie siatki,
podpisanie wykresu i obu osi

plot(real(z),imag(z));
grid;
title('Wykres funkcji z w dziedzinie liczb zespolonych');
xlabel('oś rzeczywista x');
ylabel('oś urojona y');
legend('z(x,y)')
```

Wynik działania m-pliku:



Rys. 7. Wynik działania m-pliku z Przykładu 7

Grafika trójwymiarowa – rozwinięcie

Oprócz przedstawionych wyżej poleceń związanych z grafiką dwuwymiarową w MATLAB'ie możliwe jest również rysowanie wykresów trójwymiarowych, poziomicowych i innych, szczegółowo opisanych w pomocy MATLAB'a.

Funkcja **plot3(x,y,z)** jest wersją 3-D funkcji **plot(x,y)** pozwalającą na wykreślenie projekcji linii 2-D w 3-D poprzez punkty współrzędnych wektorów x, y, z.

Przykład 8

Napisz m-plik kreślący trójwymiarowy wykres funkcji $x = \sin(z)$, $y = \cos(z)$ dla $z \in (0, 10\pi)$.

```
% Czyszczenie
clear;
clc;

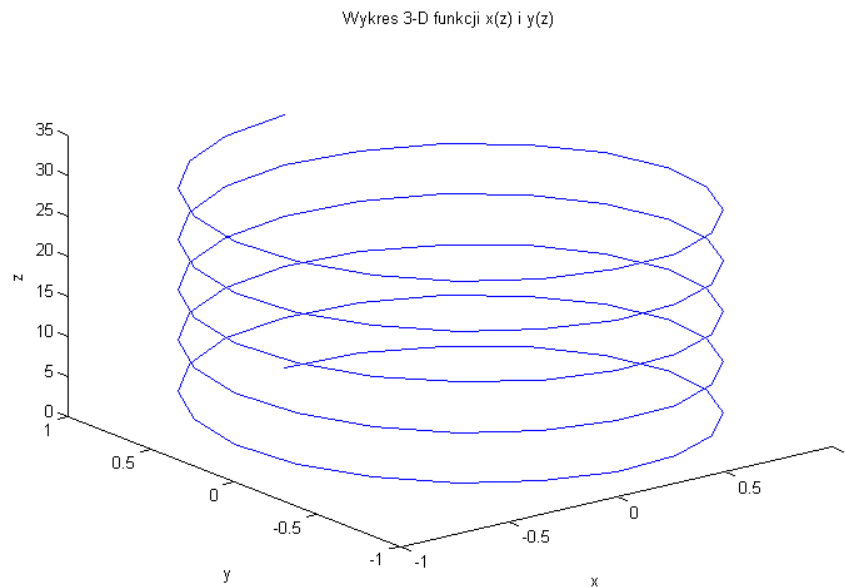
% Wyszwietlenie tekstu na ekranie
disp('M plik kreslacy wykres 3-D przy uzyciu funkcji plot3')
disp('')

% Okreslenie wartosci z
z=0:0.1*pi:10*pi;

% Obliczenie wartosci funkcji x i y
x=sin(z);
y=cos(z);

%Wykreslenie funkcji , podpisanie wykresu i osi
plot3(x,y,z);
title('Wykres 3-D funkcji x(z) i y(z)');
xlabel('x');
ylabel('y');
zlabel('z')
```

Wynik działania m-pliku:



Rys. 8. Wynik działania m-pliku z Przykładu 8

Ciekawą funkcją podobną do funkcji **plot3** jest funkcja **comet3** która wykreśla funkcję przy użyciu „głowy komety” podążającej od punktu początkowego obliczeń do punktu końcowego. Analogicznie można zastąpić wcześniej wymienioną funkcję **plot** funkcją **comet**.

Funkcje do rysowania powierzchni

Powierzchnie w programie MATLAB rysowane są wykorzystując przekształcanie wektorów x, y w macierze X, Y , tak że wiersze wynikowej macierzy X są kopiami wektora x , zaś kolumny macierzy Y są kopiami wektora y . Poleceniem które realizuje takie przekształcenie jest **meshgrid(x,y)**, funkcja **mesh(X,Y,Z)** wykreśla wynikową powierzchnię. W celu przedstawienia działania tych funkcji posłużmy się przykładem poniżej.

Przykład 9

Mając dane wektory $x=[1,2,3,4,5]$ i $y=[5,4,3,2,1]$ używając polecenia **meshgrid(x,y)** należy utworzyć macierze X i Y , następnie wyznaczyć macierz $Z = X^2 - Y^2$, którą należy wykreślić w postaci powierzchni.

```
% Czyszczenie
clear;
clc;

% Wświetlenie tekstu na ekranie
disp('M plik pokazujący działanie funkcji meshgrid')
disp('')

% Określenie wartości x i y
x=1:1:5
y=5:-1:1
```

```

% Utworzenie macierzy X i Y

[X,Y]=meshgrid(x,y)

% Obliczenie wartości funkcji
Z=X.^2-Y.^2

%Wykreslenie funkcji , podpisanie osi
mesh(X,Y,Z);
xlabel('x');
ylabel('y');
zlabel('z');

```

Wynik działania M-pliku:

x =

```

1 2 3 4 5

```

y =

```

5 4 3 2 1

```

X =

```

1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5

```

Y =

```

5 5 5 5 5
4 4 4 4 4
3 3 3 3 3
2 2 2 2 2
1 1 1 1 1

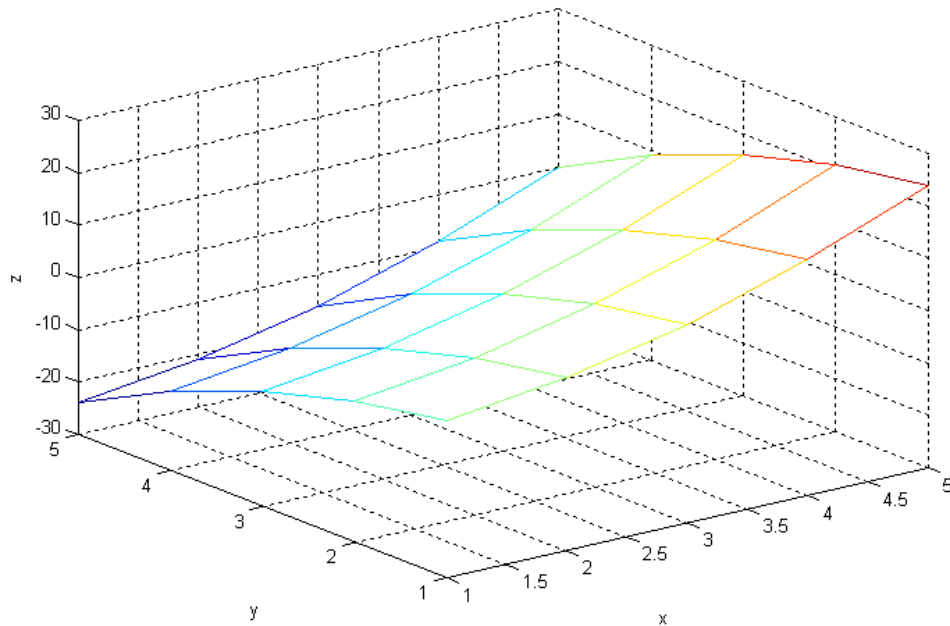
```

Z =

```

-24 -21 -16 -9 0
-15 -12 -7 0 9
-8 -5 0 7 16
-3 0 5 12 21
0 3 8 15 24

```



Rys. 9. Wynik działania m-pliku z Przykładu 9

Przykład 10

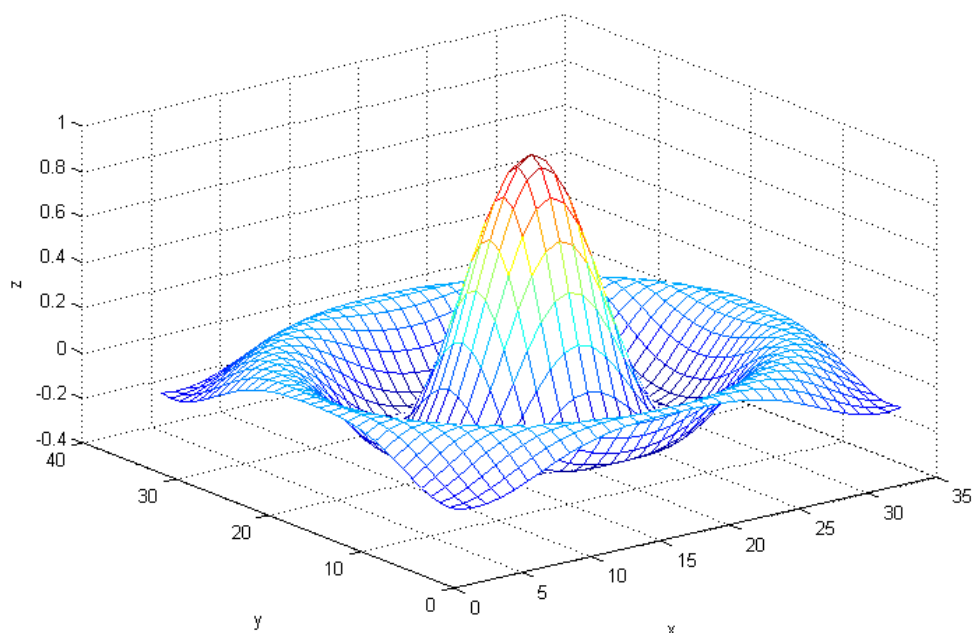
Przy użyciu tej samej metody możemy wykreślić powierzchnię przypominającą meksykańskie sombrero w postaci M-pliku:

```
% Czyszczenie
clear;
clc;

% Wświetlenie tekstu na ekranie
disp('M plik wykreslajacy meksykanskie sombrero')
disp('')

%
[X,Y]=meshgrid(-8:0.5:8)
% Obliczenie wartości funkcji
r=sqrt(X.^2+Y.^2)+eps;
z=sin(r)./r;

%Wykreslenie funkcji , podpisanie i osi
mesh(z);
xlabel('x');
ylabel('y');
zlabel('z');
```



Rys. 10. Wynik działania m-pliku z Przykładu 10.

Inne wybrane funkcje związane z grafiką trójwymiarową przedstawia tabela 5, które szczegółowo opisane są w pomocy programu MATLAB.

Tabela 5. Wybrane funkcje związane z grafiką trójwymiarową

Funkcje	Opis
contour(z)	Wykreślenie wykresu konturowego w 2-D
contour3(z)	Wykreślenie wykresu konturowego w 3-D
surf(x,y,z)	Wykreślenie wykresu powierzchniowego w 3-D
meshc(z)	Kombinacja funkcji mesh z funkcją contour.
ribbon(z), waterfall(z)	Funkcje podobne do funkcji mesh pozwalające uzyskać powierzchnie złożone odpowiednio z „wstęg” lub w kształcie „wodospadu”

Bibliografia

- Brzózka J., Dorobczyński L. *Matlab – środowisko obliczeń naukowo – technicznych*. Wydawnictwo MIKOM, 2005.
- Mrozek B., Mrozek Z. *Matlab i Simulink. Poradnik użytkownika. Wydanie II*. Wydawnictwo HELION, 2004.
- Zalewski A., Cegiela R. *Matlab – obliczenia numeryczne i ich zastosowania*. Wydawnictwo NAKOM, 1996.