



Politechnika Wrocławska

Technika obliczeniowa i symulacyjna (TOiS)





Dr inż. Andrzej Dobrucki, doc PWr

andrzej.l.dobrucki@pwr.wroc.pl

pok. 239 bud C-4



Materiały z laboratorium

www.zto.ita.pwr.wroc.pl/~ald

Index of /~ald - Windows Internet Explorer

http://zto.ita.pwr.wroc.pl/~ald/

Plik Edycja Widok Ulubione Narzędzia Pomoc

Ulubione Index of /~ald

Index of /~ald

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
Parent Directory		-	
AREK00003C/	24-Feb-2013 17:02	-	
AREK00003L/	16-Nov-2010 08:47	-	
ETEK00016L/	29-Oct-2012 09:05	-	
ETEK00022C/	24-Feb-2013 17:01	-	
ETEK00022L/	18-Sep-2011 12:13	-	
TKEK00001L/	04-Mar-2013 19:04	-	
TLEK00002C/	31-Jan-2013 07:28	-	



Witryna ZTO

www.zto.ita.pwr.wroc.pl



Instytut Telekomunikacji, Teleinformatyki i Akustyki

Zakład Teorii Obwodów

Budynek C-4 tel. i fax 0-71-320-35-29



Zakład Teorii Obwodów



M E N U

Strona główna

Działalność

Naukowa

Działalność
Dydaktyczna

Pracownicy

Dla Studentów

Poczta

Zmiana wyglądu
strony

Stronę prowadzi
dr inż. Czesław Michalik.

Zakład Teorii Obwodów

powstał w 1968 r. równocześnie z utworzeniem

Instytutu Telekomunikacji i Akustyki.

Od momentu powstania Zakładem kierował prof. dr hab. inż. Marian Piekarski.

Obecnie Zakładem kieruje dr inż. Czesław Michalik.

Polecamy:

Znakomity skrypt.
Ukazało się już drugie
wydanie.

Włodzisław Walski

**Teoretyczne
podstawy
techniki
analogowej**

Wydawnictwo Politechniki Wrocławskiej

Strona została
odwiedzona

2254

razy od
1 września 2009 r.



Dla studentów



Instytut Telekomunikacji, Teleinformatyki i Akustyki
Zakład Teorii Obwodów
Budynek C-4 tel. i fax 0-71-320-35-29



M E N U

Strona główna

Działalność

Naukowa

Działalność

Dydaktyczna

Pracownicy

Dla Studentów

Poczta

Zmiana wyglądu
strony

Stronę prowadzi
dr inż. Czesław Michalik.

Materiały Dydaktyczne

Aby zobaczyć tę stronę wymagana jest autoryzacja.

login:

hasło:

☒ Zapamiętaj hasło na czas tej sesji.

Login

Uwaga: Twoja przeglądarka musi akceptować pliki COOKIE



Polecamy:

Znakomity skrypt.
Ukazało się już drugie
wydanie.

Włodzisław Wolski

**Teoretyczne
podstawy
techniki
analogowej**

Wydawnictwo Naukowe PWN

Strona została
odwiedzona

2254

razy od
1 września 2009 r.



Technika obliczeniowa i symulacyjna

Katedra Teorii Sygnałów, Zespół Teorii Obwodów - Windows Internet Explorer

http://zto.ita.pwr.wroc.pl/index.php?strona=dla_studentow&materiale=dydaktyczne

Plik Edycja Wzrost Ułubione Narzędzia Pomoc

Ułubione Katedra Teorii Sygnałów, Zespół Teorii Obwodów

MENU

- Strona główna
- Działalność Naukowa
- Działalność Dydaktyczna
- Pracownicy
- Dla Studentów
- Zmiana wyglądu strony

Stronę prowadzi:
dr inż. Czesław Michałik.

KOMUNIKATY

	Różne ważne ogłoszenia, terminy wprowadzeń do laboratoriów, zmiany rozkładów zajęć, 'poszukiwani' studenci, itp...
	Terminy oraz wyniki egzaminów i zaliczeń Terminy konsultacji

WYKŁADY

	Technika Analogowa, dr Cz. Michałik
	Technika Analogowa, dr L. Dereń
	Technika Analogowa, dr Z. Świętch
	Technika Obliczeniowa, dr Z. Świętch
	Technika Obliczeniowa i Symulacyjna, dr Z. Świętch
	Podstawy Elektrotechniki i Elektroniki, dr A. Jarząbek

ĆWICZENIA

	Technika Analogowa
	Technika Analogowa Filia Jelenia Góra
	Podstawy Elektrotechniki i Elektroniki
	Technika Analogowa 2, kurs dodatkowy
	Teoria Obwodów

LABORATORIA

	Technika Analogowa, Podstawy Elektrotechniki i Elektroniki
	Technika Analogowa Filia Jelenia Góra
	Technika Obliczeniowa
	Technika Obliczeniowa i Symulacyjna

INNE

	Przydatne linki
--	-----------------



TOiS (wykład)

Materiały Dydaktyczne

Technika Obliczeniowa i Symulacyjna

TOS_W_1.PDF	Program Wykładu 1
TOS_W_2.PDF	Program Wykładu 2
TOS_W_3.PDF	Program Wykładu 3
TOS_W_4.PDF	Program Wykładu 4
TOS_W_5.PDF	Program Wykładu 5
TOS_W_6.PDF	Program Wykładu 6
TOS_W_7.PDF	Program Wykładu 7
TOS_W_L.PDF	Literatura
TOS_W_Z.PDF	Kolokwium zaliczeniowe, przykładowe pytania i zadania

Wstecz



TOiS (laboratorium)

Technika Obliczeniowa i Symulacyjna Laboratorium

TOS_L_1.PDF	1 Matlab I, Podstawy Programowania
TOS_L_2.PDF	2 Matlab II, Definiowanie i Wykreślanie Sygnałów
TOS_L_3.PDF	3 Matlab III, Interpolacja i Aproksymacja
TOS_L_4.PDF	4 Pspice I, Podstawy Obsługi
TOS_L_5.PDF	5 Pspice II, Metoda Symboliczna
TOS_L_6.PDF	6 Pspice III, Analiza Częstotliwościowa Filtrów Elektrycznych
TOS_L_7.PDF	7 Pspice IV, Analiza Stałoprądowa
TOS_L_8.PDF	8 Pspice V, Analiza Czasowa, Stany Nieustalone

Materiały dodatkowe i uzupełniające

NS.PDF	Nagłówek Sprawozdania
TOS_E_1.PDF	Materiały do ćwiczenia nr 1
TOS_E_2.PDF	Materiały do ćwiczenia nr 2
TOS_E_3.PDF	Materiały do ćwiczenia nr 3
PSEX.ZIP	Materiały do ćwiczenia nr 4, przykłady
TOS_E_5.PDF	Materiały do ćwiczenia nr 5
TOS_E_6.PDF	Materiały do ćwiczenia nr 6
TOS_E_7.PDF	Materiały do ćwiczenia nr 7
MT1.ZIP	Podstawy Matlaba, kilka różnych opracowań
PS1.ZIP	Podstawy Pspice, wersja 8
PS2.ZIP	Podstawy Pspice, wersja 9
IE754.ZIP	Biblioteka Matlaba, Arytmetyka Standardu IEEE 754



Politechnika Wrocławska

**Wprowadzenie
Język Matlab
Podstawowe operacje**

Typy i nazwy zmiennych

- MATLAB operuje tylko na jednym typie danych - **na macierzach**
- Wektory i skalary są szczególnymi przypadkami macierzy, tzn. posiadającymi jeden wiersz lub/i jedną kolumnę
- Nazwy zmiennych rozpoczynają się od litery, a następnie może wystąpić dowolna kombinacja liter, cyfr i znaków podkreślenia
- Pamiętanych jest 19 pierwszych znaków

POLECENIA

- ' Po wydaniu polecenia i naciśnięciu klawisza Enter Matlab natychmiast wyświetla jego wynik.
- ' Umieszczenie po poleceniu średnika spowoduje wykonanie obliczeń, ale bez zwracania wyniku.
- ' Polecenie powinno się mieścić w jednym wierszu.
- ' Kilka poleceń w jednym wierszu oddzielamy od siebie przecinkami lub średnikami.

Nazwy i definicje zmiennych

- MATLAB rozróżnia duże i małe litery w nazwach zmiennych oraz poleceń
 - polecenia standardowe należy pisać małymi literami
 - do nazywania własnych programów i zmiennych można używać małych i dużych liter
- Definiowanie typu i wielkości zmiennej odbywa się automatycznie

MATLAB – funkcje i skrypty

- MATLAB jest zintegrowanym interaktywnym środowiskiem obliczeniowym jak również i językiem programowania. Znaczy to, że można wykonywać pojedyncze komendy jak również pisać i uruchamiać programy. Bardzo istotny jest fakt, że w obu przypadkach posługujemy się dokładnie takimi samymi poleceniami.
- Pliki zawierające polecenia Matlabu nazywamy m-palikami. M-pliki można tworzyć korzystając z dowolnego edytora tekstowego, lecz najwygodniej jest posługiwać się wbudowanym wewnętrznym edytorem. Edytor wywołujemy komendą **edit**.

Ścieżka dostępu

- MATLAB wykorzystuje definicję ścieżki dostępu w celu znalezienia M-plików
- M-pliki znajdują się w odpowiednio zorganizowanych katalogach i podkatalogach
- Jeśli wprowadzimy pewną nazwę *nazwa* w linii MATLABA to interpretator linii poleceń wykona następujące czynności
 - Sprawdzi, czy *nazwa* jest zmienną
 - Sprawdzi, czy *nazwa* jest funkcją wbudowaną
 - Poszuka pliku *nazwa.m* w bieżącym katalogu
 - Przejrzy katalogi, zgodnie z stawioną ścieżką dostępu



System podpowiedzi

- Polecenie help
help magic
- Okno help
- Polecenie lookfor
lookfor inverse
- Dokumentacja na krążku lub sieci lokalnej
- Dokumentacja książkowa
- Strony WWW firmy Mathworks

Wprowadzanie danych - funkcje specjalne

- **Funkcje specjalne**
- `pi` - generuje liczbę π
- `Inf` - generuje symbol nieskończony „1/0”
- `NaN`- generuje symbol nieoznaczony „Inf/Inf” czy „0/0”
- **Uwagi.** Pojawienie się podczas obliczeń symboli „Inf”, czy „NaN” nie powoduje błędu i wartości te mogą być użyte do dalszych obliczeń. Rezultatem obliczeń z argumentem „NaN” będzie również „NaN”. „Inf” i „NaN” nie są interpretowane jako komendy graficzne.



Formaty liczb na konsoli

Format	Opis	Przykład
short	Krótki	3.1416
short e	Krótki z wykładnikiem	3.1416e+00
long	Długi	3.14159265358979
long e	Długi z wykładnikiem	3.14159265358979e+00
bank	Bardzo krótki	3.14
hex	Szesnastkowy	400921fb54442d18
+	Tylko znak liczb	++
compact	Zwarty w pionie	

Format można zmieniać poleceniem format

>>format long e



Edycja linii komend

Klawisz	Funkcja
↑, ^P	Przywołaj poprzednią linię
↓, ^N	Przywołaj następną linię
←, ^B	Przesuń kursor w lewo o jeden znak
→, ^F	Przesuń kursor w prawo o jeden znak
Delete	Usuń znak
^L	Przesuń kursor w lewo o jeden wyraz
^R	Przesuń kursor w prawo o jeden wyraz
^A	Przesuń kursor do początku linii
^E	Przesuń kursor do końca linii
^U	Usuń bieżącą linię
^T	Przełącz tryb pisania
^D	Usuń znak pod kursorem
^K	Usuń do końca linii



Definiowanie macierzy

- przez wymienianie elementów

$A = [2 \ 2 \ 2 \ 1; 1 \ 2 \ 3 \ 1]$ lub $A = [1 \ 2 \ 3 \ 4 \ \dots$

$5 \ 6 \ 7 \ 8]$ jest to równoważne

$A = [1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$

$B = []$ – macierz pusta

- przez wygenerowanie

Wykorzystujemy wyrażenie o postaci ogólnej: min:krok:max.

Generuje ono wektor wierszowy o następującej budowie

$[\text{min}, \text{min}+\text{krok}, \text{min}+2*\text{krok}, \dots, \text{max}]$. Jeśli krok zostanie pominięty to przyjmowany domyślnie jest $\text{krok}=1$.

Przykład

```
>> a=[1:10;40:2:58]
```

```
a =
```

1	2	3	4	5	6	7	8	9	10
40	42	44	46	48	50	52	54	56	58



Definiowanie macierzy

- przez zbudowanie z innych macierzy

Przykład

```
>> A=[1 4 1;2 0 1],B=[3 1;4 1],C=[1 2 2 0 1;2 4 7 1 0]
A =
     1     4     1
     2     0     1
B =
     3     1
     4     1
C =
     1     2     2     0     1
     2     4     7     1     0
```

Zbuduj macierz D postaci

$$\begin{matrix} AB \\ C \end{matrix}$$



Definiowanie macierzy

Rozwiązanie: » `D=[A B;C]`

`D =`

1	4	1	3	1
2	0	1	4	1
1	2	2	0	1
2	4	7	1	0

Gdyby » `A=[1 1;2 2],D=[A B;C]`

`A =`

1	1
2	2

??? All rows in the bracketed expression must have the same number of columns.

Wszystkie wiersze macierzy muszą mieć tę samą liczbę kolumn.

Definiowanie macierzy

- przez mieszanie omówionych wyżej technik

Przykład

```
>> A=[1 2 3;4 5 6], D=[A,[1;2];1:4]
```

A =

1	2	3
4	5	6

D =

1	2	3	1
4	5	6	2
1	2	3	4



Funkcje wspomagające konstruowaniu macierzy

eye – macierz jednostkowa

```
» eye(2), eye(2,4), eye([2 4])
```

ans =

1	0
0	1

ans =

1	0	0	0
0	1	0	0

ans =

1	0	0	0
0	1	0	0

ones- macierz o wszystkich elementach równych jeden

```
» ones(2), ones(2,4), ones([2 4])
```

ans =

1	1
1	1

ans =

1	1	1	1
1	1	1	1

ans =

1	1	1	1
1	1	1	1



Funkcje wspomagające konstruowaniu macierzy

zeros – macierz o wszystkich elementach równych zero

```
» zeros(2), zeros(2,4), zeros([2 4])
```

ans =

```
0 0
0 0
```

ans =

```
0 0 0 0
0 0 0 0
```

ans =

```
0 0 0 0
0 0 0 0
```

rand wypełnia macierz liczbami pseudolosowymi

```
» rand(5)
```

ans =

```
0.7019 0.8511 0.3722 0.1219 0.8234
0.8477 0.5620 0.0737 0.5221 0.0466
0.2093 0.3193 0.1998 0.1171 0.5979
0.4551 0.3749 0.0495 0.7699 0.9492
0.0811 0.8678 0.5667 0.3751 0.2888
```



Generowanie ciągów liczb

`linspace(x1,x2,N)` - generuje wektor wierszowy N-elementowy (gdy N opuścimy to $N=100$) liczb równomiernie rozmieszczonych między wartościami $x1$ i $x2$

`logspace(x1,x2,N)` – generuje wektor wierszowy N-elementowy liczb rozmieszczonych w sposób logarytmiczny w przedziale 10^{x1} do 10^{x2} .



Dostęp do elementów macierzy

Niech `>> A=[1 2 3 4 5 6;0 9 8 7 6 5;1 1 0 0 2 2]`

`A =`

1	2	3	4	5	6
0	9	8	7	6	5
1	1	0	0	2	2

- odwołania do elementów `>> A(2,4)`

`ans =`

Np. `7`

`>> A(3,3)`

`ans =`

`0`



Dostęp do elementów macierzy

A =

1	2	3	4	5	6
0	9	8	7	6	5
1	1	0	0	2	2

- odwoływanie do podmacierzy

» A(2,:)

ans =

0	9	8	7	6	5
---	---	---	---	---	---

» A([1 3],:)

ans =

1	2	3	4	5	6
1	1	0	0	2	2

Np.

» A(1:2,1:2)

ans =

1	2
0	9

» A(:, [1:3 5])

ans =

1	2	3	5
0	9	8	6
1	1	0	2

» A([1 3],1:2:5)

ans =

1	3	5
1	0	2



Dostęp do elementów macierzy

- traktowanie macierzy jako wektora kolumnowego

W wielu zastosowaniach przydatna może okazać się możliwość potraktowania macierzy jako długiego wektora kolumnowego, którym kolejne kolumny umieszczono jedna pod drugą.

Niech

```
» A=magic(3)
```

```
A =
```

```
8     1     6
3     5     7
4     9     2
```

```
» A(5)
```

```
ans =
```

```
5
```

```
» A(:)
```

```
ans =
```

```
8
3
4
1
5
9
6
7
2
```




Rozmiar macierzy i ich wyświetlanie

```
a =
```

```
    1    2  
    3    4  
    5    6
```

```
» [n,m]=size(a)
```

```
n =
```

```
    3
```

```
m =
```

```
    2
```



Macierze puste i ich usuwanie

$A = [1 \ 2 \ 3 \ 4; 4 \ 5 \ 6 \ 7];$

$A(:,4) = []$ – usuwa ostatnią (4) kolumnę macierzy A

$A(:,1:2) = []$ – usuwa dwie pierwsze kolumny macierzy A

Operatory macierzowe – suma, różnica i transpozycja

$A+B$ – suma,

$A-B$ – różnica,

$A*B$ – iloczyn,

A' – transpozycja,

Operatory i wyrażenia tablicowe (operacje z kropką)

$A.*B$ – iloczyn element * element,

$A./B$ – iloraz,

$A.^n$ – potęgowanie

Operacje na tablicowe na macierzach

- Dostępne są operacje $.*$ $./$ $.\backslash$ $.^$, które wykonuje się element po elemencie macierzy
- Macierze muszą być tych samych wymiarów
- Dodawanie i odejmowanie tablicowe są zdefiniowane tak samo jak dla macierzy.



Mnożenie tablicowe

```
F-Secure SSH - DEFAULTS.SSH
File Edit View Help
[Icons]
>> a=[1 2 -3; 4 -5 6; -7 8 9];
>> b=[-1 2 3; 4 -5 6; 7 8 -9];
>> c=a.*b

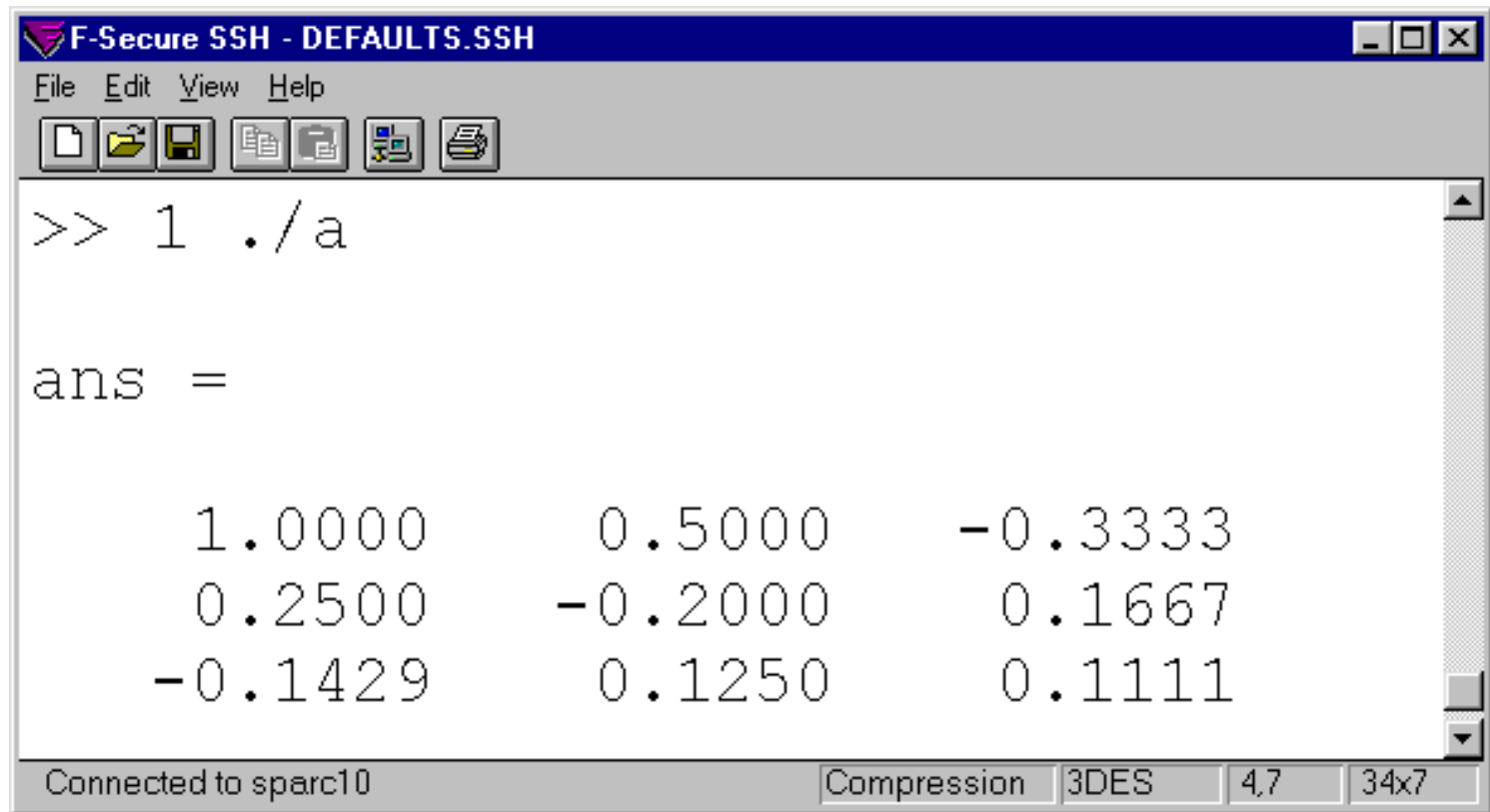
C =

    -1     4    -9
    16    25    36
   -49    64   -81

>> 
```

Connected to sparc10 Compression 3DES 4,11 35x11

Mnożenie i dzielenie przez skalar



The screenshot shows a terminal window titled "F-Secure SSH - DEFAULTS.SSH". The prompt is ">>". The user has entered "1 ./a". The output shows the result of a scalar multiplication of a 3x3 matrix by the scalar 1. The result is displayed as a 3x3 matrix of floating-point numbers.

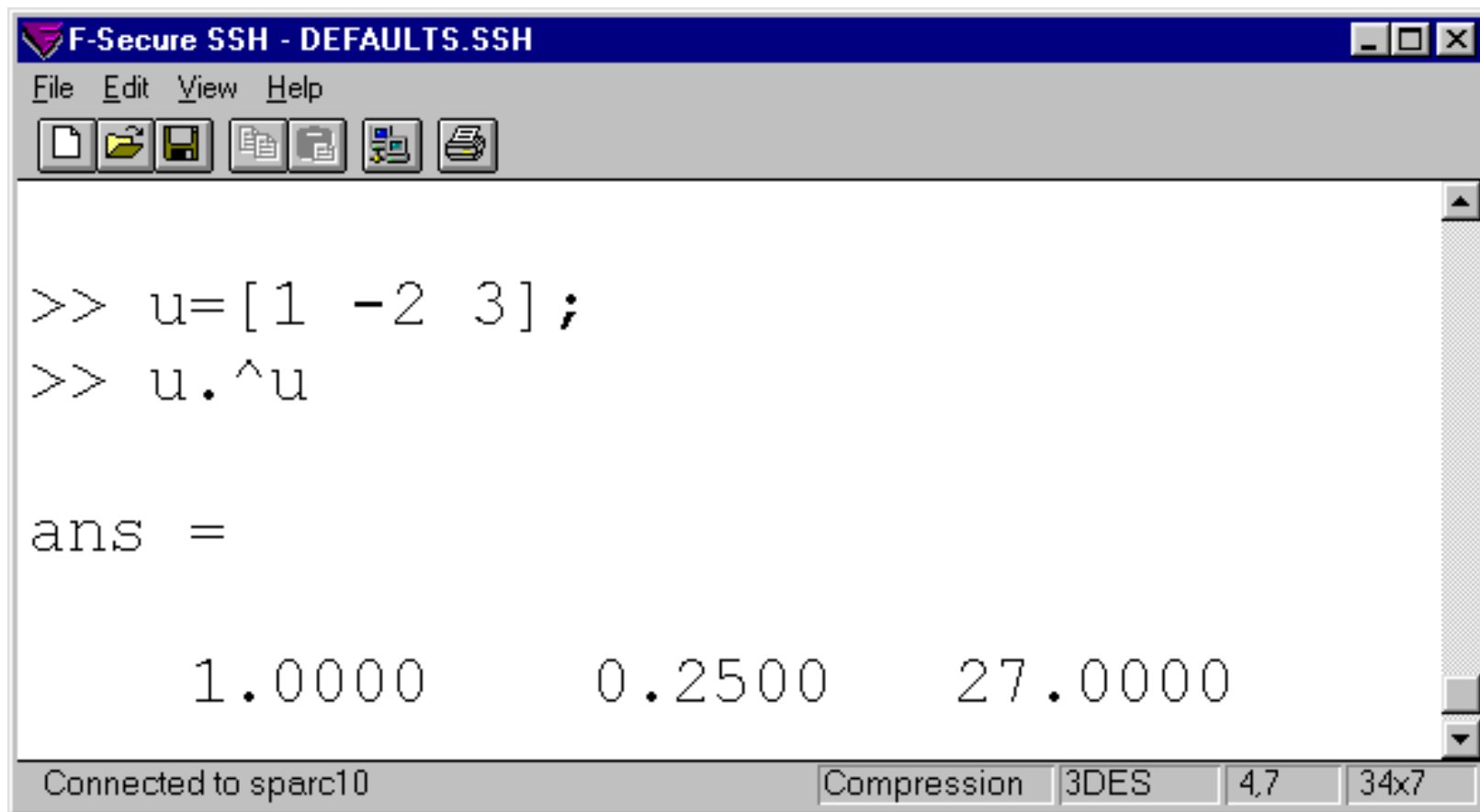
```
>> 1 ./a

ans =

    1.0000    0.5000   -0.3333
    0.2500   -0.2000    0.1667
   -0.1429    0.1250    0.1111
```

At the bottom of the window, it says "Connected to sparc10". On the right side, there are settings for "Compression" (set to "3DES"), "4,7", and "34x7".

Potęgowanie tablicowe .^



The screenshot shows a terminal window titled "F-Secure SSH - DEFAULTS.SSH". The window has a menu bar with "File", "Edit", "View", and "Help", and a toolbar with icons for file operations. The terminal content shows a MATLAB session where a vector u is defined as $[1 \ -2 \ 3]$, and then the element-wise power $u.^u$ is calculated. The result is displayed as a row vector: $1.0000 \quad 0.2500 \quad 27.0000$. The status bar at the bottom indicates "Connected to sparc10" and shows settings for "Compression", "3DES", "4.7", and "34x7".

```
F-Secure SSH - DEFAULTS.SSH
File Edit View Help
[Icons]

>> u=[1 -2 3];
>> u.^u

ans =

    1.0000    0.2500   27.0000

Connected to sparc10  Compression  3DES  4.7  34x7
```

Potęgowanie tablicowe .^ (skalary)

```
F-Secure SSH - DEFAULTS.SSH
File Edit View Help
[Icons]

>> a.^2

ans =

     1     4     9
    16    25    36
    49    64    81

Connec Compression 3DES 4.7 18x7
```

```
F-Secure SSH - DEFAULTS.SSH
File Edit View Help
[Icons]

>> 2 .^a

ans =

    2.0000    4.0000    0.1250
   16.0000    0.0312   64.0000
    0.0078  256.0000  512.0000

Connected to sparc10 Compression 3DES 4.7 30x7
```



Funkcje logiczne badające własności elementów macierzy

```
>> X=[0 0 2;0 5 -1]
```

X =

```
0    0    2
0    5   -1
```

```
>> I=find(X)
```

I =

```
4
5
6
```

```
>> I=find(X>1)
```

I =

```
4
5
```

```
>> [I,J]=find(X>1)
```

I =

```
2
1
```

J =

```
2
3
```

Zmienić kolejność kolumn lub wierszy w macierzy

[m,n] = size(A)

Q = A(m:-1:1,:) - odwraca kolejność wierszy,

W = A(:,n:-1:1) – odwraca kolejność kolumn



Rozwiązywanie układu równań liniowych w Matlabie

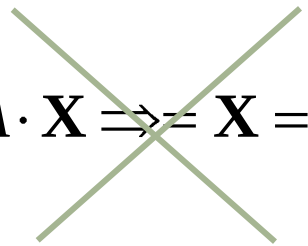
$$d_1 = a_{11}x_1 + a_{12}x_2$$

$$d_2 = a_{21}x_1 + a_{22}x_2$$

Zadane wartości: x_1, x_2, y_1, y_2

Należy znaleźć: x_1, x_2


$$\begin{bmatrix} d_1 \\ d_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$


$$\mathbf{D} = \mathbf{A} \cdot \mathbf{X} \Rightarrow \mathbf{X} = \mathbf{A}^{-1} \cdot \mathbf{D}$$


$$\mathbf{D} = \mathbf{A} \cdot \mathbf{X}, \mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \mathbf{D} = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$$

`A=[a11 a12;a21 a22];`

`X=[x1;x2];`

`X=A\D;`

`x1=X(1);x2=X(2);`



Fragment skryptu w Matlabie



Podsumowanie

Operacje na macierzach

Definiowanie macierzy:	Przykłady:
przez wymienienie elementów	$A = [2 \ 3 \ 4 \ 6; 1 \ 2 \ 3 \ 4]$ lub $A = [2 \ 3 \ 4 \ 6$ $1 \ 2 \ 3 \ 4]$ $B = [3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 8 \ 9]$
przez zbudowanie z innych macierzy	mając zdefiniowane macierze: $A = [1 \ 4 \ 1; 2 \ 0 \ 1];$ $B = [3 \ 1; 4 \ 1];$ $C = [1 \ 2 \ 2 \ 0 \ 1; 2 \ 4 \ 7 \ 1 \ 0]$ po wydaniu polecenia: $D = [A \ B; C]$ otrzymamy nową macierz:
przez wygenerowanie elementów wykorzystując ogólne wyrażenie: <u>minimum:krok:maximum</u>	wydanie polecenia: $A = [1:5; 1:2:9]$ spowoduje wygenerowanie macierzy: $A = [1 \ 2 \ 3 \ 4 \ 5]$ $1 \ 3 \ 5 \ 7 \ 9]$
przez usuwanie wektora z macierzy	mając macierz $A = [1 \ 2 \ 3 \ 4; 4 \ 5 \ 6 \ 7]$ i wydając polecenie: $A(:, 4) = []$ spowodujemy usunięcie 4-ej kolumny macierzy A, natomiast polecenie: $A(:, 1:2) = []$ usunie dwie pierwsze kolumny macierzy A



Funkcje ułatwiające konstruowanie macierzy

Funkcja	Opis
eye(n)	tworzy macierz jednostkową o wymiarze nxn (jedyńki na przekątnej)
eye(m,n)	tworzy macierz jednostkową o wymiarze mxn np: X=eye(size(X))
ones(n)	tworzy macierz o wymiarze nxn o wszystkich elementach równych 1
ones(m,n)	tworzy macierz o wymiarze mxn o wszystkich elementach równych 1
zeros(n)	tworzy macierz o wymiarze nxn o wszystkich elementach równych 0
zeros(m,n)	tworzy macierz o wymiarze mxn o wszystkich elementach równych 0
rand(n)	tworzy macierz o wymiarze nxn wypełnioną liczbami pseudolosowymi z przedziału <0, 1>
rand(m,n)	tworzy macierz o wymiarze mxn wypełnioną liczbami pseudolosowymi z przedziału <0, 1>



Wyznaczenie rozmiarów macierzy

Funkcja	Opis
disp(A)	wysyła macierz A do okna poleceń Matlaba
[n,m]=size(A)	zwraca liczbę wierszy n i liczbę kolumn m macierzy A
n=size(A,1)	zwraca liczbę wierszy n macierzy A
m=size(A,2)	zwraca liczbę kolumn macierzy A
n=length(X)	podaje wymiar n wektora X (dla macierzy zwraca dłuższy z jej wymiarów)



Funkcje przekształcające macierze

Układy równań liniowych

$Y = \text{rot90}(X)$	obrócenie macierzy X o 90 stopni w kierunku przeciwnym do ruchu wskazówek zegara.
$Y = \text{reshape}(X, n, m)$	utworzenie macierzy Y o n wierszach i m kolumnach z elementów branych kolejno kolumnami z macierzy X . Jeśli X nie zawiera $m \times n$ elementów to pojawia się komunikat o błędzie.
$[L, U] = \text{lu}(A)$	dokonyje rozkładu LU macierzy A tak że $A = L * U$, gdzie L - macierz trójkątna dolna, U - macierz trójkątna górna
$x = \text{inv}(A) * b$ lub $x = A \backslash b$	rozwiązanie układu równań $Ax = b$ (b musi być wektorem kolumnowym)
$x = b / A$	rozwiązanie układu równań $xA = b$ (b musi być wektorem wierszowym)

Funkcje trygonometryczne

- $\sin$ - sinus
- $\cos$ - cosinus
- $\tan$ - tangens
- $\arcsin$ - arcus sinus
- $\arccos$ - arcus cosinus
- $\arctan$ - arcus tangens
- $\arctan2$ - arcus tangens w czterech ćwiartkach
- $\sinh$ - sinus hiperboliczny
- $\cosh$ - cosinus hiperboliczny
- $\tanh$ - tangens hiperboliczny
- $\operatorname{arcsinh}$ - arcus sinus hiperboliczny
- $\operatorname{arccosh}$ - arcus cosinus hiperboliczny
- artanh - arcus tangens hiperboliczny



Funkcje elementarne

- abs - wartość bezwzględna
- angle - faza
- unwrap - (w radianach), eliminuje skoki fazy o 2π
- conj - sprzężenie
- imag - część urojona
- real - część rzeczywista
- fix - zaokrąglenie w kierunku 0
- floor - zaokrąglenie w kierunku $-\infty$
- ceil - zaokrąglenie w kierunku $+\infty$
- round - zaokrąglenie do najbliższej całkowitej
- rem - reszta z dzielenia
- sign - znak
- exp - funkcja wykładnicza
- log - logarytm naturalny
- log10 - logarytm dziesiętny
- sqrt - pierwiastek kwadratowy

Wielomiany

- roots - pierwiastki wielomianu
- poly - wielomian na podstawie pierwiastków
- polyval - wartość wielomianu
- residue - rozkład na ułamki proste
- polyfit - dopasowanie wielomianu do danych
- polyder - pochodna wielomianu
- conv - mnożenie wielomianów
- deconv - dzielenie wielomianów

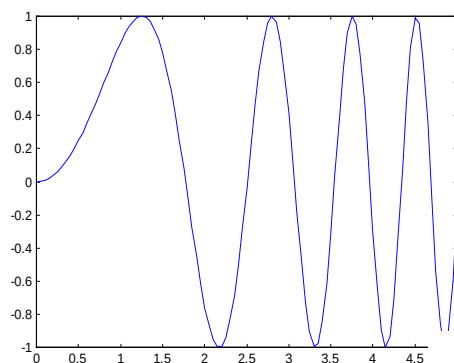
Operacje podstawowe i interpolacja

- max - element maksymalny
- min - element minimalny
- mean - średnia
- median - mediana
- std - odchylenie standardowe
- sort - sortowanie w porządku rosnącym
- sum - suma
- prod - iloczyn

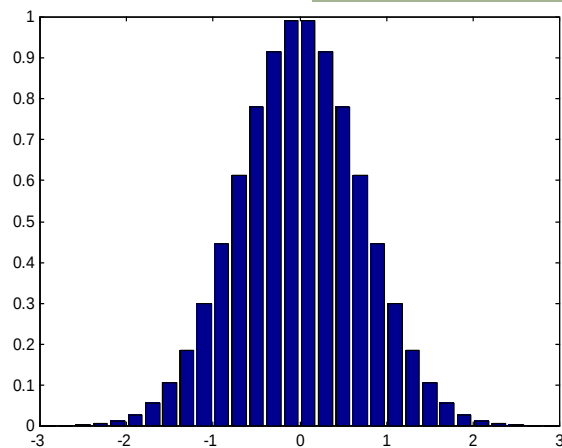


MATLAB – wykresy 2D

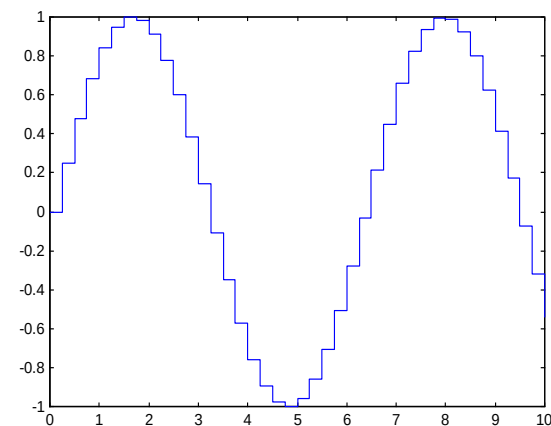
plot



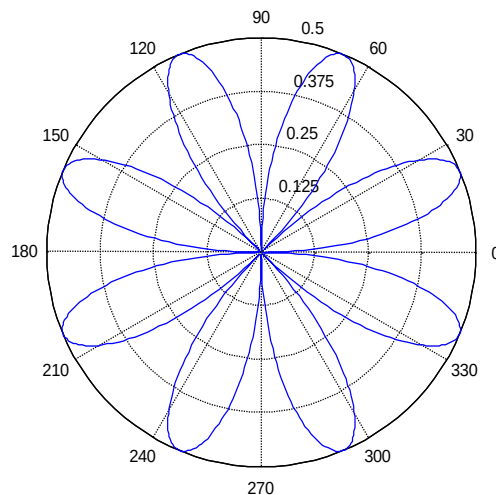
bar



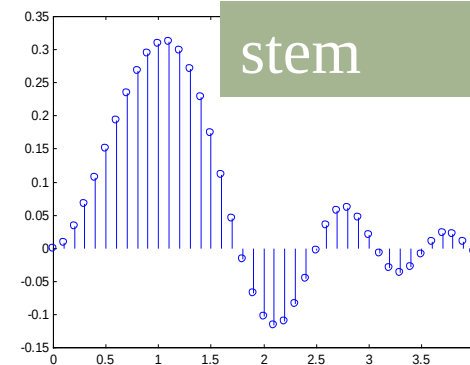
stairs



polar



stem





MATLAB – wykresy 2D

- `figure` lub `figure(n)`
- `plot(X,Y)`
- `plot(X,Y,S)`
- `plot(X1,Y1,X2,Y2,X3,Y3,...)`
- `plot(X1,Y1,S1,X2,Y2,S2,X3,Y3,S3,...)`
- `semilogx` - j.w. oś X logarytmiczna
- `semilogy` - j.w. oś Y logarytmiczna
- `loglog` - j.w. obie osie logarytmiczne
- `polar(THETA, RHO)` - współrzędne biegunowe, kąt, moduł, nie można zadawać wiele par danych
- `zplane(Z,P)` - wykreśla zera Z i bieguny P (wektory kolumnowe)
- Inne polecenia - `bar`, `stairs`, `stem`

Znaczenie znaków w łańcuchu S

- Znaczenie znaków w łańcuchu S

- | | | | | | |
|-----|---------|-----|------------------|----|---------|
| • y | yellow | • . | point | - | solid |
| • m | magenta | • o | circle | : | dotted |
| • c | cyan | • x | x-mark | -. | dashdot |
| • r | red | • + | plus | -- | dashed |
| • g | green | • * | star | | |
| • b | blue | • s | square | | |
| • w | white | • d | diamond | | |
| • k | black | • v | triangle (down) | | |
| • | | • ^ | triangle (up) | | |
| • | | • < | triangle (left) | | |
| • | | • > | triangle (right) | | |
| • | | • p | pentagram | | |
| • | | • h | hexagram | | |
- plot(X,Y,'r+:')
 - Wykres koloru czerwonego, punkty zaznaczone „plusikami” i połączone kropkami

MATLAB – funkcja HOLD

**Zatrzymuje
bieżący wykres w
okienku**

**hold on
hold off
hold**

**Funkcja hold określa czy nowo
rysowany wykres zostanie dodany
do bieżącego wykresu, czy go
zastąpi.**

**hold on zatrzymuje bieżący
wykres i pewne właściwości osi
tak że następne polecenia
rysowania dodają kolejne
wykresy do bieżącego.**

MATLAB – funkcja SUBPLOT

Tworzy i kontroluje wykresy o wielu osiach

subplot(m,n,p)

Subplot dzieli bieżące okienko wykresu na kwadratowe obszary, które są numerowane rzędami. Każdy obszar zawiera osie.

subplot(m,n,p) tworzy osie w p-tym obszarze okienka podzielonego na m-na-n kwadratowych obszarów. Nowe osie stają się osiami bieżącymi.

1	2	3
4	5	6

MATLAB – funkcja GRID

**Rysuje linie
siatki dla
wykresów
2-D i 3-D**

**grid on
grid off
grid**

**Funkcja grid włącza i wyłącza linie
siatki dla bieżących osi.**

**grid on dodaje siatkę dla bieżących
osi.**

**grid off usuwa siatkę dla bieżących
osi.**

**grid przełącza stan widoczności
siatki.**

MATLAB – funkcja TITLE

Dodaje opis do bieżącej osi

```
title('string')  
title(fname)  
title(...,'PropertyName',...  
.PropertyValue,...)
```

Każdy obiekt axes może posiadać jeden opis (tytuł).

Opis jest zlokalizowany pośrodku, powyżej osi.

title('string') wyświetla string pośrodku, powyżej bieżącej osi.

title(fname) wywołuje funkcję zwracającą łańcuch tekstowy który jest wyświetlany pośrodku powyżej bieżącej osi.

MATLAB – funkcja XLABEL, YLABEL, ZLABEL

**Opis osi
x, y ,z**

```
xlabel('string')  
xlabel(fname)  
xlabel(...,  
 'PropertyName'  
,...PropertyName'  
e,...)
```

Każdy obiekt axes może posiadać jeden opis dla osi x, y, z. Opis pojawia się poniżej odpowiedniej osi w przypadku wykresu 2-D i obok lub poniżej w przypadku wykresu 3-D.

xlabel('string') opisuje oś x dla bieżących osi.

xlabel(fname) wywołuje funkcję zwracającą łańcuch tekstowy który jest wyświetlany pośrodku powyżej osi x dla bieżącego układu osi.

xlabel(...,'PropertyName',PropertyValue,...) określa wartości (PropertyValue) dla danej własności (PropertyName) dla obiektu graficznego Text utworzonego przez xlabel.

α	<code>\alpha</code>	θ	<code>\theta</code>	τ	<code>\tau</code>
β	<code>\beta</code>	π	<code>\pi</code>	υ	<code>\upsilon</code>
γ	<code>\gamma</code>	ϕ	<code>\phi</code>	δ	<code>\delta</code>
κ	<code>\kappa</code>	ρ	<code>\rho</code>	λ	<code>\lambda</code>
χ	<code>\chi</code>	ε	<code>\varepsilon</code>	μ	<code>\mu</code>
ψ	<code>\psi</code>	ζ	<code>\zeta</code>	ν	<code>\nu</code>
σ	<code>\sigma</code>	ω	<code>\omega</code>	η	<code>\eta</code>
ξ	<code>\xi</code>	Γ	<code>\Gamma</code>	Λ	<code>\Lambda</code>
Σ	<code>\Sigma</code>	Ψ	<code>\Psi</code>	Δ	<code>\Delta</code>
Ξ	<code>\Xi</code>	Υ	<code>\Upsilon</code>	Ω	<code>\Omega</code>
Θ	<code>\Theta</code>	Π	<code>\Pi</code>	Φ	<code>\Phi</code>
$\times$	<code>\times</code>	$\pm$	<code>\pm</code>	$\div$	<code>\div</code>
$\leq$	<code>\leq</code>	$\geq$	<code>\geq</code>	$\equiv$	<code>\equiv</code>
$\approx$	<code>\approx</code>	$\cong$	<code>\cong</code>	$\neq$	<code>\neq</code>

MATLAB – funkcje GRID, AXIS, LEGEND

- `grid on`
- `grid off`
- `grid`
- Funkcja `grid` powoduje wyświetlenie na wykresie „siatki”
- `axis([XMIN XMAX YMIN YMAX])` - pozwala zmienić skalę aktywnego wykresu, argumentem jest czteroelementowy wektor
- `legend(string1,string2,string3, ...)` - umieszcza na wykresie legendę
- `legend(string1 ,...,pos)` - umieszcza w wyznaczonym miejscu