

Kinematyka i Dynamika Robotów

Instrukcja do zajęć laboratoryjnych

dr inż. Dawid Pietrala

dpietrala@tu.kielce.pl

1 Laboratorium 1: Wprowadzenie do Wolfram Mathematica

1.1 Cele zajęć

Celem laboratorium jest zapoznanie studentów z podstawami pracy w środowisku *Wolfram Mathematica* oraz z elementami języka *Wolfram Language*, które będą wykorzystywane w dalszej części kursu z kinematyki i dynamiki robotów.

W trakcie zajęć studenci nauczą się:

- wykonywać podstawowe obliczenia w Mathematica,
- definiować wektory i macierze,
- wykonywać operacje macierzowe,
- tworzyć wykresy funkcji,
- generować wykresy na podstawie list danych,
- wizualizować wektory w przestrzeni 3D,
- tworzyć proste animacje.

Zagadnienia te będą stanowić podstawę do dalszych ćwiczeń związanych z kinematyką robotów manipulacyjnych.

1.2 Wymagane środowisko

Do realizacji ćwiczeń konieczne jest posiadanie:

- Wolfram Mathematica (wersja 12 lub nowsza).

Zajęcia będą wykonywane w pliku typu *Notebook* (.nb).

Uruchomienie komórki z kodem następuje za pomocą skrótów:

Shift + Enter

1.3 Podstawy języka Wolfram Language

W Mathematica wszystkie operacje wykonywane są w postaci wyrażeń symbolicznych.

Przykładowe obliczenia:

```
2 + 3
5^2
Sin[Pi/4]
```

Definiowanie zmiennych:

```
a = 5
b = 3
a + b
```

Definicja funkcji:

```
f[x_] := x^2 + 2 x + 1
f[3]
```

Funkcje matematyczne zapisujemy z dużej litery, np.

Sin, Cos, Tan

1.4 Ćwiczenie 1: Operacje na wektorach

W robotyce często wykonujemy operacje na wektorach w przestrzeni trójwymiarowej. W Mathematica wektory zapisujemy jako listy.

```
v1 = {1, 2, 3}
v2 = {-1, 0, 4}
```

Dodawanie wektorów:

```
v1 + v2
```

Mnożenie przez skalar:

```
3 v1
```

Iloczyn skalarny:

```
v1.v2
```

Norma wektora:

```
Norm[v1]
```

Normalizacja wektora:

```
Normalize[v1]
```

1.5 Ćwiczenie 2: Operacje na macierzach

Macierze zapisujemy jako listy list.

```
A = {
  {1, 2, 3},
  {0, 1, 4},
  {5, 6, 0}
}
```

Wyświetlenie macierzy w czytelnej formie:

```
MatrixForm[A]
```

Transpozycja macierzy:

```
Transpose[A]
```

Wyznacznik macierzy:

```
Det[A]
```

Macierz odwrotna:

```
Inverse[A]
```

1.6 Mnożenie macierzy

Mnożenie macierzy realizowane jest za pomocą operatora kropki.

```
A = {
  {1, 2},
  {3, 4}
};
```

```
B = {
  {0, 1},
  {1, 0}
};
```

```
A.B
```

Możliwe jest również mnożenie macierzy przez wektor:

```
v = {5, 6};
A.v
```

Operacja ta będzie bardzo często wykorzystywana w robotyce, między innymi do składania transformacji oraz transformacji punktów w przestrzeni.

1.7 Ćwiczenie 3: Wizualizacja wektorów w przestrzeni 3D

Mathematica umożliwia łatwe tworzenie wizualizacji w przestrzeni trójwymiarowej.

```
v1 = {1, 2, 1};
v2 = {2, -1, 2};
```

```
Graphics3D[
  {
    Red, Arrow[{{0,0,0}, v1}],
    Blue, Arrow[{{0,0,0}, v2}]
  },
  Axes -> True
]
```

Dodanie etykiet:

```
Graphics3D[
  {
    Red, Arrow[{{0,0,0}, v1}],
    Blue, Arrow[{{0,0,0}, v2}],
    Text["v1", v1],
    Text["v2", v2]
  },
  Axes -> True
]
```

1.8 Ćwiczenie 4: Wykresy funkcji

Mathematica posiada rozbudowane funkcje do generowania wykresów.

```
Plot[Sin[x], {x, 0, 2 Pi}]
```

Wykres parametryczny:

```
ParametricPlot[
  {Cos[t], Sin[t]},
  {t, 0, 2 Pi}
]
```

Wykres powierzchni w przestrzeni 3D:

```
Plot3D[
  Sin[x] Cos[y],
  {x, -Pi, Pi},
```

```
{y, -Pi, Pi}
]
```

1.9 Ćwiczenie 5: Generowanie danych za pomocą Table

W Mathematica bardzo często potrzebujemy wygenerować listę danych na podstawie określonego wzoru matematycznego. Do tego celu służy funkcja `Table[]`.

Najprostszy przykład generowania listy liczb:

```
Table[i, {i, 1, 10}]
```

Polecenie to generuje listę kolejnych liczb od 1 do 10.
Możliwe jest również generowanie wartości funkcji.

```
Table[Sin[i], {i, 0, 10}]
```

Często przydatne jest określenie kroku zmiany zmiennej:

```
Table[t, {t, 0, 1, 0.1}]
```

W ten sposób można wygenerować listę punktów opisujących funkcję.

```
Table[{t, Sin[t]}, {t, 0, 2 Pi, 0.1}]
```

Otrzymujemy listę par punktów (x, y) , które można bezpośrednio wykorzystać do rysowania wykresów.

Przykład wykorzystania funkcji `Table[]` do wygenerowania trajektorii punktu:

```
traj = Table[{t, Sin[t]}, {t, 0, 2 Pi, 0.1}]
```

Lista ta może być następnie użyta do wizualizacji przy pomocy funkcji `ListLinePlot[]`.

1.10 Ćwiczenie 6: Wykresy z list danych

W praktyce często posiadamy dane w postaci listy próbek. Do ich wizualizacji służy funkcja `ListLinePlot[]`.

```
dane = {0, 1, 0.5, 2, 1.5, 1.8, 1.2};
```

```
ListLinePlot[dane]
```

Można również podać jawnie pary punktów:

```
punkty = {
  {0, 0},
  {1, 1},
  {2, 0.5},
  {3, 2},
  {4, 1.2}
};
```

```
ListLinePlot[punkty]
```

Na jednym wykresie można przedstawić kilka przebiegów:

```
traj1 = Table[{t, Sin[t]}, {t, 0, 2 Pi, 0.1}];
traj2 = Table[{t, Cos[t]}, {t, 0, 2 Pi, 0.1}];
```

```
ListLinePlot[{traj1, traj2}]
```

1.11 Ćwiczenie 7: Prosta animacja

Mathematica umożliwia tworzenie animacji bez dodatkowych bibliotek.

```
Animate[
  Graphics[
    {
      Arrow[{{0,0}, {Cos[t], Sin[t]}}]
    },
    Axes -> True,
    PlotRange -> 1.5
  ],
  {t, 0, 2 Pi}
]
```

1.12 Zadania do samodzielnego rozwiązania

Poniższe zadania mają na celu utrwalenie materiału omawianego w trakcie zajęć. W każdym zadaniu należy przygotować kod w środowisku Mathematica.

1.12.1 Zadania podstawowe

Zadanie 1: Operacje na wektorach

Zdefiniuj dwa wektory trójwymiarowe:

```
a = {1,2,3}
b = {3,-1,2}
```

Następnie oblicz:

- sumę wektorów,
- różnicę wektorów,
- iloczyn skalarny,
- normę obu wektorów,
- wektor znormalizowany.

Zadanie 2: Iloczyn wektorowy

Zdefiniuj dwa dowolne wektory trójwymiarowe i oblicz ich iloczyn wektorowy za pomocą funkcji `Cross[]`.

Sprawdź również, czy otrzymany wektor jest prostopadły do obu wektorów wejściowych.

Zadanie 3: Operacje na macierzach

Zdefiniuj macierz

```
A = {
  {1,2,3},
  {4,5,6},
  {7,8,9}
}
```

Następnie oblicz:

- transpozycję macierzy,
- wyznacznik,
- macierz A^2 ,
- iloczyn macierzy A przez wektor $\{1, 1, 1\}$.

Zadanie 4: Mnożenie macierzy

Zdefiniuj dwie macierze:

```
A = {{1,2},{3,4}}
B = {{0,1},{1,0}}
```

Oblicz:

- iloczyn AB ,
- iloczyn BA .

Sprawdź, czy wyniki są identyczne.

Zadanie 5: Wykres funkcji

Narysuj wykres funkcji

$$f(x) = \sin(x) + 0.5 \cos(2x)$$

dla zakresu

$$x \in [0, 4\pi]$$

za pomocą funkcji `Plot[]`.**Zadanie 6: Wykres z list danych**

Wygeneruj listę danych

$$y_k = \sin(0.2k)$$

dla $k = 0, 1, \dots, 50$ i narysuj wykres przy użyciu funkcji `ListLinePlot[]`.**1.12.2 Zadania rozszerzone****Zadanie 7: Wizualizacja wektorów**

Narysuj w przestrzeni 3D trzy wektory:

```
a = {1,0,1}
b = {0,2,1}
c = {-1,1,2}
```

Każdy wektor powinien mieć inny kolor oraz etykietę.

Następnie oblicz wektor

$$w = a - 2b + c$$

i również przedstaw go na tym samym wykresie.

Zadanie 8: Trajektoria parametryczna

Narysuj trajektorię opisaną równaniami:

$$x(t) = \cos(t)$$

$$y(t) = \sin(2t)$$

dla

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

za pomocą funkcji `ParametricPlot[]`.**Zadanie 9: Trajektoria w przestrzeni 3D**

Narysuj krzywą parametryczną w przestrzeni:

$$x(t) = \cos(t)$$

$$y(t) = \sin(t)$$

$$z(t) = 0.2t$$

dla

$$t \in [0, 10\pi]$$

Użyj funkcji `ParametricPlot3D[]`.

Zadanie 10: Animacja wektora

Utwórz animację wektora obracającego się w płaszczyźnie XY:

$$v(t) = \begin{bmatrix} \cos(t) \\ \sin(t) \end{bmatrix}$$

Wektor powinien być narysowany jako strzałka wychodząca z początku układu współrzędnych.

Zadanie 11: Animacja układu dwóch wektorów

Zdefiniuj dwa wektory

$$a = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Następnie przygotuj animację, w której wektor a obraca się o kąt t , a wektor b pozostaje nieruchomy.

Wskazówka: użyj funkcji `RotationMatrix[]`.

Zadanie 12: Wykres wielu trajektorii

Wygeneruj trzy przebiegi:

$$y_1(t) = \sin(t)$$

$$y_2(t) = \cos(t)$$

$$y_3(t) = \sin(2t)$$

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

Narysuj wszystkie trzy przebiegi na jednym wykresie przy użyciu `ListLinePlot[]`.