

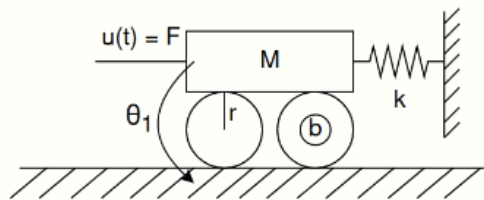
Sprawozdanie z projektu

Metody Modelowania Matematycznego

Szymon Kowalski 198055 ACiR2B

Maciej Kotowski 198001 ACiR2B

Projekt 7. Dany jest model wózka na sprężynie z zamontowanym tłumikiem na kołach:



Należy wyprowadzić model układu oraz zaimplementować go w symulacji. Symulator powinien umożliwiać pobudzenie układu przynajmniej trzema rodzajami sygnałów wejściowych (prostokątny o skończonym czasie trwania, trójkątny, harmoniczny). Symulator powinien umożliwiać zmianę wszystkich parametrów układu oraz sygnałów wejściowych. Należy użyć metody Rungego-Kutty 4-go rzędu oraz metody Eulera oraz na wspólnym wykresie pokazać wyniki symulacji (położenia i prędkości wózka) z obu tych metod.

Oznaczenia symboli:

M - masa wózka

k - współczynnik sprężystości sprężyny

b - współczynnik tłumienia (obrotowy)

J - moment bezwładności koła

r - promień koła

x - położenie wózka

\dot{x} - prędkość wózka

\ddot{x} - przyspieszenie wózka

θ - kąt obrotu

$\dot{\theta}$ - prędkość kątowna

$\ddot{\theta}$ - przyspieszenie kątowe

Model Stanowy:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{k}{\frac{J}{r^2} - M} & \frac{-b}{r^1(\frac{J}{r^2} - M)} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{-1}{\frac{J}{r^2} - M} \end{bmatrix} u(t)$$

Realizacja:

Projekt został zrealizowany w pythonie z wykorzystaniem bibliotek:

- tkinter - GUI
- numpy - obliczenia numeryczne
- matplotlib - rysowanie wykresów
- scipy - do utworzenia sygnałów wejściowych

Projekt został podzielony na 4 pliki źródłowe:

- main.py - GUI oraz rysowanie wykresów
- rk4.py - obliczenia numeryczne metodą Rungego-Kutty 4-go rzędu
- euler.py - obliczenia numeryczne metodą eulera
- signal_generator.py - generacja sygnałów wejściowych

Do wykonania projektu wykorzystaliśmy IDE PyCharm Professional oraz GIT

Interfejs Graficzny:

Umożliwia:

- zmianę parametrów modelu (k , J , r , M , b)
- zmianę parametrów sygnału wejściowego i jego kształt
- zmianę parametrów symulacji (czas trwania symulacji oraz długość kroku)

cart-modeling-simulation

Enter sim values:

Const k	10
Const J	0.5
Const r	2
Const M	1
Const b	0.1
Discretization time const	0.001
Simulation time	10

Input parameters:

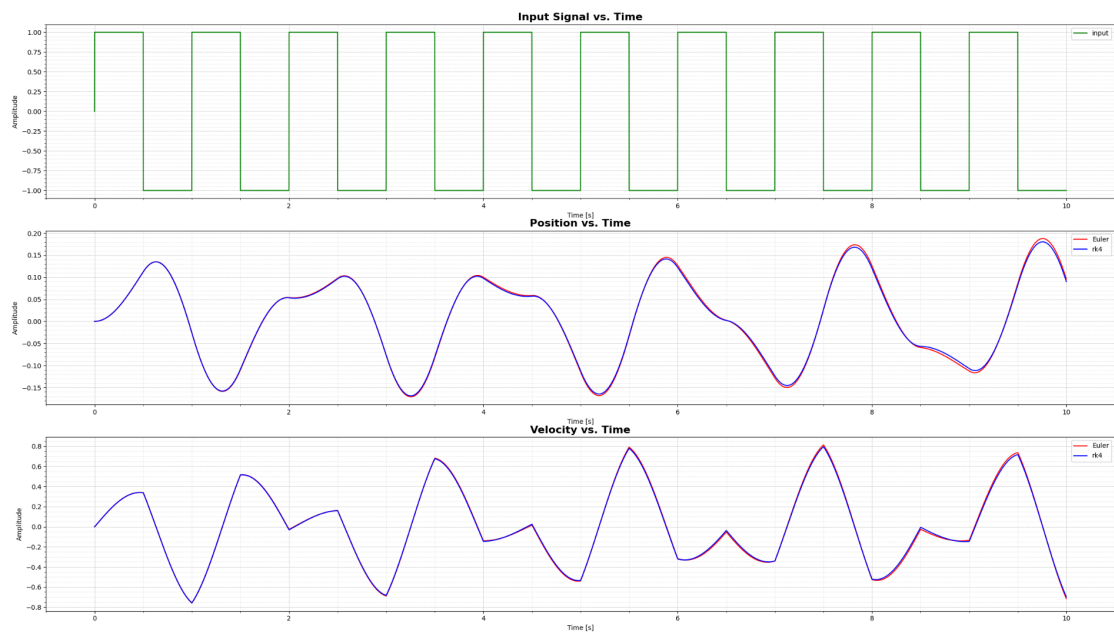
Amplitude	1
Frequency (Hz)	1

Signal type:

☒ rectangular ☐ triangular ☐ harmonic

Start

Przykładowe wyniki symulacji:



Wnioski:

- Metoda Rungego-Kutty 4-go rzędu jest bardziej precyzyjna
- Metoda Eulera wymaga bardzo małego kroku czasowego, aby uzyskać akceptowalną dokładność.
- Metoda Rungego-Kutty 4-go zachowuje większą precyzję przy zwiększaniu kroku czasowego symulacji niż metoda eulera