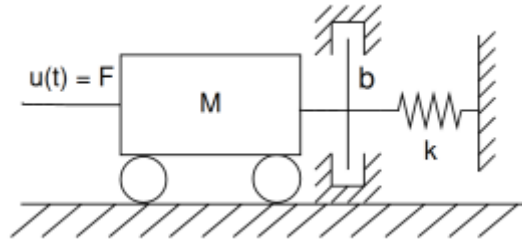


Projekt z przedmiotu Metody Modelowania Matematycznego
Adrian Nowogrodzki 184332
Hubert Boratyński 184648

Opis działania

Program pozwala na symulowanie działania wózka zaczepionego na sprężynie wraz z tłumieniem, zgodnie z poniższym schematem:

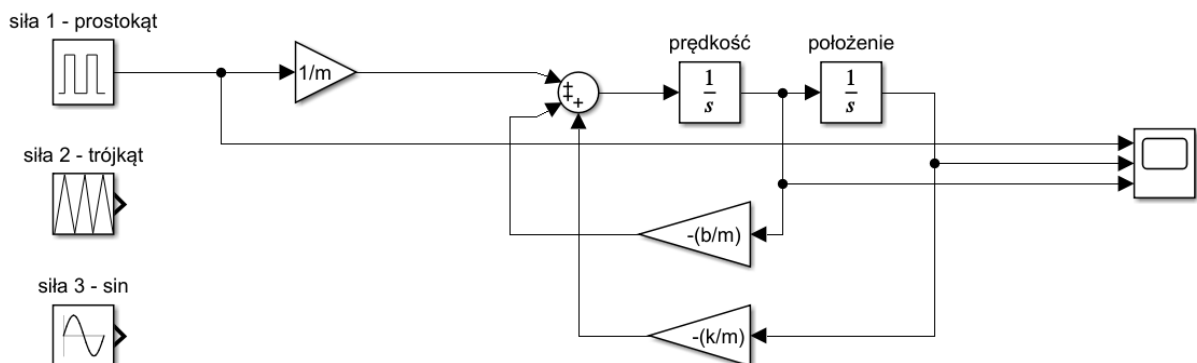


Wielkością wejściową w układzie jest siła, natomiast wyjściem jest położenie oraz prędkość wózka. Użytkownik ma możliwość wyboru metody symulacji, zmiany parametrów układu, ustalenia szczegółów przebiegu symulacji, a następnie wyświetlenia sygnału wejściowego (siły) oraz sygnałów wyjściowych (prędkości oraz położenia) na wspólnym wykresie. Program napisany został w języku Python z wykorzystaniem biblioteki *tkinter* do stworzenia graficznego interfejsu użytkownika.

Model stanowy układu:

$$\begin{bmatrix} x' \\ v' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{bmatrix} * \begin{bmatrix} x \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} * f$$

Schemat blokowy układu:



Metoda Eulera:

$$x_{n+1} = x_n + hf(t_n, x_n), \quad t_n = t_0 + nh,$$

W metodzie Eulera przybliżamy wartość kolejnego punktu używając bezpośrednio wartości punktu poprzedniego oraz jego pochodnej pomnożonej przez krok programu.

Metoda Rungego Kutty 4 rzędu:

$$\bar{\mathbf{a}}_n = \bar{\mathbf{f}}(\bar{\mathbf{x}}_n)$$

$$\bar{\mathbf{b}}_n = \bar{\mathbf{f}}(\bar{\mathbf{x}}_n + \frac{h}{2} \bar{\mathbf{a}}_n)$$

$$\bar{\mathbf{c}}_n = \bar{\mathbf{f}}(\bar{\mathbf{x}}_n + \frac{h}{2} \bar{\mathbf{b}}_n)$$

$$\bar{\mathbf{d}}_n = \bar{\mathbf{f}}(\bar{\mathbf{x}}_n + h \bar{\mathbf{c}}_n)$$

$$\bar{\mathbf{x}}_{n+1} = \bar{\mathbf{x}}_n + \frac{h}{6} (\bar{\mathbf{a}}_n + 2 \bar{\mathbf{b}}_n + 2 \bar{\mathbf{c}}_n + \bar{\mathbf{d}}_n)$$

W metodzie Rungego Kutty przybliżamy wartość kolejnego punktu używając wartości poprzedniego punktu oraz pochodnych o zmienionych parametrach.

Implementacja metody Eulera i Rungego Kutty 4 rzędu w programie:

```
# pierwsza próbka Euler i RK-4
if time == 0:
    x_now_eu = x0
    v_now_eu = v0
    x_now_rk = x0
    v_now_rk = v0
else:
    # kolejne próbki Euler
    x_now_eu = x_before_eu + step * v_before_eu
    v_now_eu = v_before_eu + step * ((1/m)*force - (k/m)*x_before_eu - (b/m)*v_before_eu)

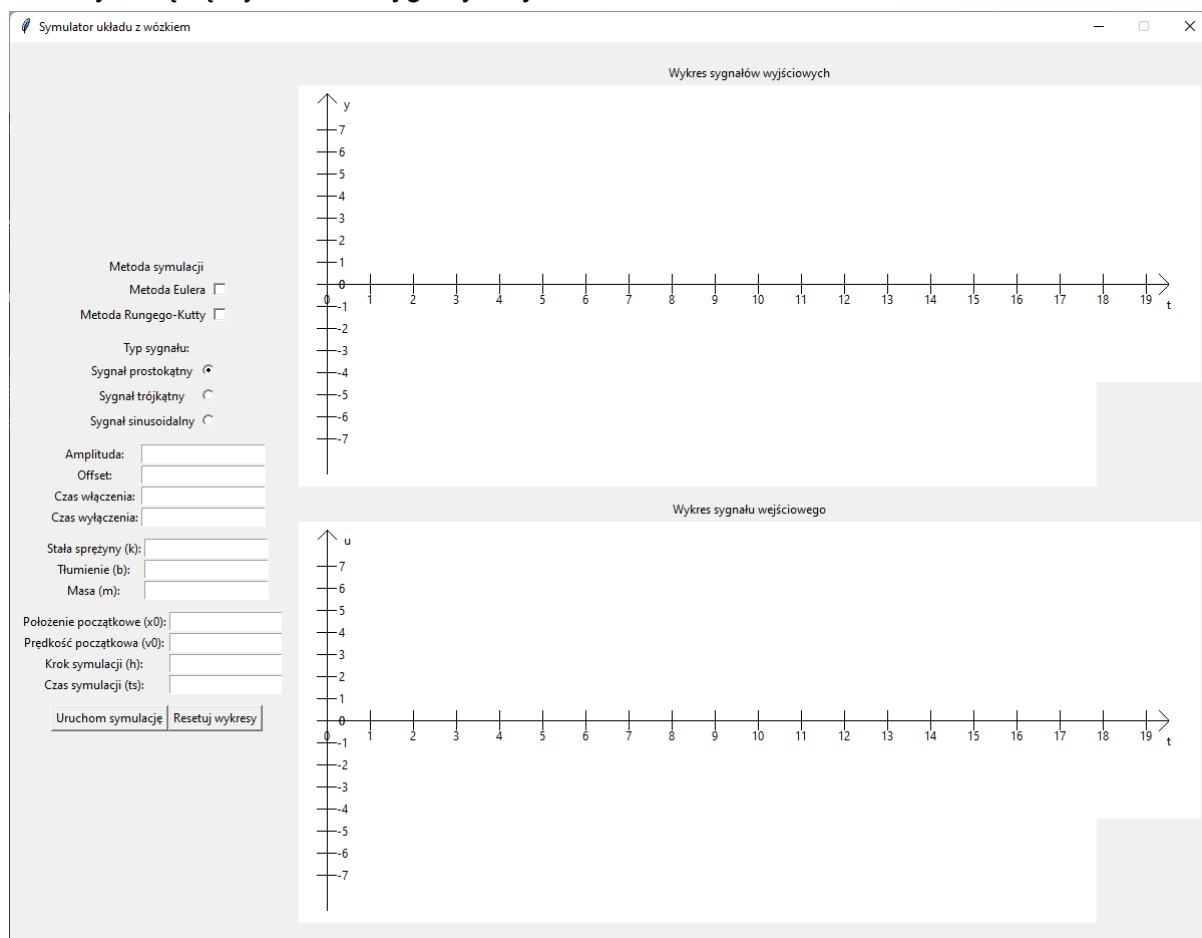
    # kolejne próbki RK-4
    a1 = v_before_rk
    b1 = v_before_rk + step/2 * a1
    c1 = v_before_rk + step/2 * b1
    d1 = v_before_rk + step * c1
    x_now_rk = x_before_rk + step/6 * (a1+2*b1+2*c1+d1)

    a2 = (1/m)*force - (k/m)*x_before_rk - (b/m)*v_before_rk
    b2 = (1/m)*force - (k/m)*(x_before_rk + step/2*a2) - (b/m)*(v_before_rk + step/2*a2)
    c2 = (1/m)*force - (k/m)*(x_before_rk + step/2*b2) - (b/m)*(v_before_rk + step/2*b2)
    d2 = (1/m)*force - (k/m)*(x_before_rk + step *c2) - (b/m)*(v_before_rk + step *c2)
    v_now_rk = v_before_rk + step/6 * (a2+2*b2+2*c2+d2)

    # przygotowania do kolejnego kroku
    time = time + step
    x_before_eu = x_now_eu
    v_before_eu = v_now_eu
    x_before_rk = x_now_rk
    v_before_rk = v_now_rk
```

Interfejs

Okno graficznego interfejsu użytkownika programu składa się z dwóch sekcji: z lewej strony znajdują się pola do wprowadzania danych przez użytkownika, a z prawej strony – wykresy, na których będą wykresiane sygnały z symulatora.



Parametry, które można ustawiać w symulatorze:

- **Metoda symulacji.** Dostępna jest metoda Eulera oraz Rungego-Kutty czwartego rzędu. Można wybrać tylko jedną z tych metod lub obie jednocześnie – w tym drugim przypadku przebiegi z obydwu metod zostaną narysowane na wspólnym wykresie.
- **Typ sygnału.** Do wyboru są sygnały: prostokątny, trójkątny oraz sinusoidalny. W zależności od wybranego typu sygnału zmianie będą ulegać możliwe do wpisania parametry sygnału. Mogą być to:
 - amplituda sygnału,
 - offset pionowy lub składowa stała,
 - czas włączenia i wyłączenia (dla impulsu prostokątnego)
 - okres (dla sygnału trójkątnego i sinusoidalnego)
 - opóźnienie sygnału na osi czasu (sygnał trójkątny)
 - faza początkowa (dla sygnału sinusoidalnego)
- **Parametry modelu.** W wyprowadzonym modelu matematycznym układu znajdują się trzy parametry, które można dostrajać: sztywność sprężyny k , stałą tłumienia b , oraz masę wózka m .
- **Parametry symulacji.** Dwa pierwsze pola edycji umożliwiają zadanie warunków początkowych dla zmiennych stanu w symulacji, którymi są położenie oraz prędkość wózka. W następnym polu wpisuje się długość kroku symulacji h , czyli czas, co który

wykonywane będzie wyznaczenie nowych wartości zmiennych stanu w modelu. Ostatnie pole odpowiada za wybór czasu symulacji t_s . Domyślnie symulacja rozpoczyna się w chwili $t_0 = 0$, więc wartość wpisana w tym okienku będzie stanowić wyznacznik zakończenia symulacji – nastąpi ona dla $t = t_s$.

Wszystkie parametry wpisywane są w jednostkach układu SI.

Poniżej pól do edycji znajdują się dwa przyciski:

- Uruchom symulację. Ten przycisk powoduje pobranie wartości z lewej części interfejsu od użytkownika, wykonanie walidacji danych i, w przypadku braku zastrzeżeń, wykonanie symulacji modelu oraz wykreślenie przebiegów wyjściowych.
- Resetuj wykresy. Użycie przywraca wykresy do stanu początkowego.

Po prawej stronie w interfejsie widoczne są dwa wykresy.

- Wykres sygnału wejściowego. Stanowi podgląd zadanego przez użytkownika sygnału, zgodnie ze wpisanymi parametrami.
- Wykres sygnałów wyjściowych. Tutaj wyświetlane będą sygnały wyjściowe z symulatora.

Oba wykresy posiadają mechanizm automatycznego dostosowania liczby działek na każdej osi oraz skalowania w taki sposób, aby przebiegi zmieściły się w okienku, w którym są rysowane. Ponadto w prawym dolnym rogu znajduje się legenda, która także jest automatycznie uzupełniana po wykreśleniu sygnałów. Umieszczanie danych odbywa się w postaci nanoszenia punktów na wykres – stanowi to dodatkowe źródło informacji dla użytkownika o dokładności symulacji. Przykładowo, jeżeli wybrany krok symulacji jest względnie duży, to wykres nie będzie przypominać linii ciągłej, lecz będą widoczne wyraźnie wyodrębnione punkty bez interpolacji wartości między nimi.

Wartości na wykresach należy interpretować w jednostkach układu SI.

Na poniższym zrzucie ekranu zaprezentowane zostały wyniki przykładowej symulacji wykonanej w programie:

