

Wahadło odwrócone

KRZYSZTOF PIEKORZ
PIOTR ZAJĄC

| | | |
|-------|---|----|
| 1 | Spis treści | |
| 2 | Wstęp..... | 2 |
| 3 | Model matematyczny wahadła matematycznego..... | 2 |
| 3.1 | Założenia | 2 |
| 3.2 | Wyprowadzenie modelu matematycznego | 2 |
| 3.2.1 | Równanie ruchu w poziomie..... | 3 |
| 3.2.2 | Równanie ruchu w pionie | 5 |
| 3.2.3 | Równanie dla ruchu obrotowego | 5 |
| 3.2.4 | Równania ruchu wahadła odwróconego | 6 |
| 3.2.5 | Linearyzacja w punkcie pracy..... | 7 |
| 3.2.6 | Reprezentacja macierzowa | 9 |
| 4 | Identyfikacja parametrów modelu | 9 |
| 4.1 | Pozyskanie danych do identyfikacji z rzeczywistego modelu | 9 |
| 4.2 | Stworzenie obiektu wahadła w Matlab | 11 |
| 4.3 | Identyfikacja parametrów modelu nieliniowego..... | 12 |
| 5 | Algorytm sterujący | 15 |
| 5.1 | Cele algorytmu i założenia | 15 |
| 5.2 | Normalizacja kąta wychylenia wahadła | 15 |
| 5.3 | Ograniczenie położenia wózka..... | 15 |
| 5.4 | Rozbijanie wahadła | 16 |
| 5.5 | Przełączanie kontrolerów stabilizującego i wychylającego | 17 |
| 5.6 | Regulator stabilizujący | 17 |
| 5.6.1 | Regulator LQ | 17 |
| 5.6.2 | Regulator PID | 18 |
| 5.7 | Saturacja i kompensacja tarcia | 19 |
| 6 | Symulacja stabilizacji wahadła | 19 |
| 6.1 | Stabilizacja w otoczeniu punktu pracy..... | 19 |
| 6.2 | Podniesienie wahadła i stabilizacja..... | 20 |
| 6.3 | Symulacja stabilizacji wahadła dla modelu nieliniowego | 21 |
| 7 | Wnioski..... | 24 |

2 Wstęp

Celem realizowanego projektu w ramach laboratorium problemowego było zapoznanie się z wybranym układem sterowania, a następnie przeprowadzenie niezbędnych kroków w celu opracowania strategii i algorytmu sterowania. Proces wdrażania regulatora dla obiektu sterowania obejmuje modelowanie obiektu, identyfikację parametrów modelu, dobór struktury i nastaw regulatora dla wybranego zadania regulacji, symulacje komputerowe oraz eksperymenty z rzeczywistym obiektem, które nie mogły zostać przeprowadzone i zostały zastąpione eksperymentami symulacyjnymi.

Do analizy i modelowania został wybrany układ wahadła odwróconego na wózku. Jest to jeden z najprostszych problemów, a jednocześnie poruszający wiele zagadnień z zakresu automatyki i sterowania, z którymi należy się zapoznać, aby skutecznie zaimplementować algorytm sterujący. Problem wahadła odwróconego na wózku polega na doprowadzeniu pręta do kąta wychylenia takiego, że „stoi” on pionowo na wózku. Jest to nietrywialne zadanie, ponieważ przy braku sterowania i kompensacji minimalne zakłócenie wprowadzono do układu powoduje wypadnięcie wahadła z tego punktu równowagi. Dodatkowo należy doprowadzić wahadło do pozycji zadanej, czyli „rozbujać”, co wiąże się z koniecznością dokładnego przestudiowania dynamiki obiektu i wykorzystaniem jej własności w celu jak najszybszego zrealizowania postawionego zadania.

3 Model matematyczny wahadła matematycznego

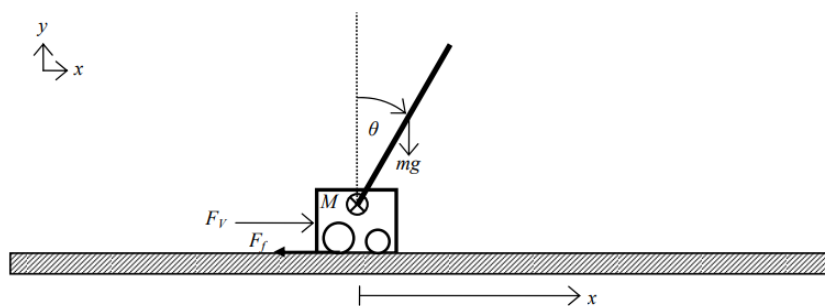
3.1 Założenia

Przed przystąpieniem do modelowania systemu wahadła matematycznego przyjęto kilka założeń:

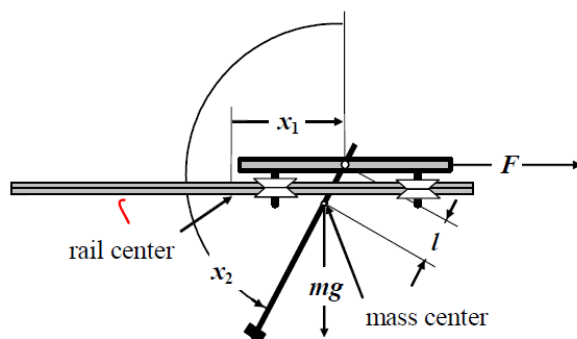
1. Wahadło potraktujemy jako jednorodny pręt, o długości l (model nieliniowy);
2. Wahadło traktujemy jako punkt masowy, oddalony od osi obrotu o odległość l (model liniowy);

3.2 Wyprowadzenie modelu matematycznego

Rysunki wahadła odwróconego:



Rys. 1 Wahadło odwrócone - pozycja stabilizacji



Rys. 2 Wahadło odwrócone - pozycja startowa

Gdzie:

$x_1 = x$ – odległość wózka od środka toru (punktu równowagi)

$x_2 = \theta$ – odchył wahadła od pozycji pionowej w górę, zgodnie z kierunkiem dodatnim matematycznie

F – siła przyłożona do wózka przez napęd,

l – odległość środka ciężkości wahadła od punktu od wózka.

3.2.1 Równanie ruchu w poziomie

Zgodnie z drugą zasadą dynamiki Newtona, ruch poziomy wózka można zapisać w postaci:

$$M \cdot \frac{d^2 x}{dt^2} = F - F_x - F_f, \quad (1)$$

Gdzie:

M – masa wózka,

F_x – siła działająca na wahadło,

F_f – siła tarcia wózka, dana jest ona wzorem:

$$F_f = \gamma_2 \frac{dx}{dt}. \quad (2)$$

Po podstawieniu siły tarcia, otrzymujemy:

$$M \cdot \frac{d^2 x}{dt^2} = F - F_x - \gamma_2 \frac{dx}{dt}. \quad (3)$$

Ruch poziomy wahadła można zapisać następująco:

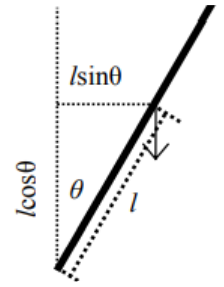
$$F_x = m \frac{d^2 x_G}{dt^2}. \quad (4)$$

Para (x_G, y_G) to współrzędne środka ciężkości wahadła i można je wyrazić z użyciem x i θ :

$$\begin{aligned} x_G &= x + l \sin \theta, \\ y_G &= l \cos \theta \end{aligned} \quad (5)$$

W celu policzenia drugiej pochodnej $(\frac{d^2 x_G}{dt^2})$, należy podstawić x_G i policzyć najpierw pierwszą pochodną:

$$\begin{aligned} \frac{dx_G}{dt} &= \frac{d(x + l \sin \theta)}{dt} = \frac{dx}{dt} + l \frac{d(\sin \theta)}{dt} \\ &= \frac{dx}{dt} + l \cos \theta \frac{d\theta}{dt} \end{aligned} \quad (6)$$



W ostatnim przejściu wykorzystany został wzór na pochodną funkcji złożonej. Następnie można policzyć drugą pochodną:

$$\begin{aligned} \frac{d^2 x_G}{dt^2} &= \frac{d}{dt} \left(\frac{dx}{dt} + l \cos \theta \frac{d\theta}{dt} \right) = \frac{d^2 x}{dt^2} + l \frac{d}{dt} \left(\cos \theta \frac{d\theta}{dt} \right) \\ &= \frac{d^2 x}{dt^2} + l \left(\frac{d \cos \theta}{dt} \frac{d\theta}{dt} + \cos \theta \frac{d^2 \theta}{dt^2} \right) \\ &= \frac{d^2 x}{dt^2} + l \left(-\sin \theta \left(\frac{d\theta}{dt} \right)^2 + \cos \theta \frac{d^2 \theta}{dt^2} \right) \\ &= \frac{d^2 x}{dt^2} - l \sin \theta \left(\frac{d\theta}{dt} \right)^2 + l \cos \theta \frac{d^2 \theta}{dt^2} \end{aligned} \quad (7)$$

W powyższych przekształceniach zostały zastosowane kolejno wzory na pochodną sumy, pochodną iloczynu i pochodną funkcji złożonej.

Obliczony wzór można podstawić do równania ruchu i wygląda ono następująco:

$$\begin{aligned} M \cdot \frac{d^2 x}{dt^2} &= F - m \left(\frac{d^2 x}{dt^2} - l \sin \theta \left(\frac{d\theta}{dt} \right)^2 + l \cos \theta \frac{d^2 \theta}{dt^2} \right) - \gamma_2 \frac{dx}{dt} \\ M \cdot \frac{d^2 x}{dt^2} &= F - m \frac{d^2 x}{dt^2} + m l \sin \theta \left(\frac{d\theta}{dt} \right)^2 - m l \cos \theta \frac{d^2 \theta}{dt^2} - \gamma_2 \frac{dx}{dt}. \end{aligned} \quad (8)$$

Po przeniesieniu pochodnych dx/dt na lewą stronę:

$$(M + m) \frac{d^2x}{dt^2} + \gamma_2 \frac{dx}{dt} = F + m l \sin \theta \left(\frac{d\theta}{dt} \right)^2 - m l \cos \theta \frac{d^2\theta}{dt^2}. \quad (9)$$

3.2.2 Równanie ruchu w pionie

Zgodnie z drugą zasadą dynamiki Newtona, równanie ruchu w pionie dla wahadła można zapisać następująco:

$$F_Y - mg = m \frac{d^2y_G}{dt^2}, \quad (10)$$

Obliczając kolejno pierwszą i drugą pochodną (5), otrzymujemy (zgodnie ze wzorem na pochodną funkcji złożonej i pochodną iloczynu):

$$\begin{aligned} \frac{dy_G}{dt} &= \frac{d(l \cos \theta)}{dt} = -l \sin \theta \frac{d\theta}{dt}, \\ \frac{d^2y_G}{dt^2} &= \frac{d}{dt} \left(-l \sin \theta \frac{d\theta}{dt} \right) = -l \left(\frac{d \sin \theta}{dt} \frac{d\theta}{dt} + \sin \theta \frac{d^2\theta}{dt^2} \right) \\ &= -l \left(\cos \theta \left(\frac{d\theta}{dt} \right)^2 + \sin \theta \frac{d^2\theta}{dt^2} \right) \\ &= -l \cos \theta \left(\frac{d\theta}{dt} \right)^2 - l \sin \theta \frac{d^2\theta}{dt^2}. \end{aligned} \quad (11)$$

Obliczoną pochodną można podstawić do równania ruchu. Będzie ono postaci:

$$F_Y = mg + m \left(-l \cos \theta \left(\frac{d\theta}{dt} \right)^2 - l \sin \theta \frac{d^2\theta}{dt^2} \right). \quad (12)$$

3.2.3 Równanie dla ruchu obrotowego

Zgodnie z drugą zasadą dynamiki Newtona dla ruchu obrotowego, zależność między momentem siły i przyspieszeniem kątowym obiektu dana jest następująco:

$$\vec{M} = I \frac{d^2\theta}{dt^2}, \quad (13)$$

gdzie \vec{M} to wypadkowy moment siły działający na wahadło.

Sumując moment siły działający na wahadło w zależności od kąta wychylenia otrzymujemy zależność:

$$F_Y l \sin \theta - F_X l \cos \theta = I \frac{d^2 \theta}{dt^2}, \quad (14)$$

gdzie I to moment bezwładności wahadła.

3.2.4 Równania ruchu wahadła odwróconego

Znając wzór na F_Y (12) i F_X (4) można podstawić do powyższego wzoru i dokonać przekształceń algebraicznych w celu uproszczenia postaci:

$$\begin{aligned} & \left(mg + m \left(-l \cos \theta \left(\frac{d\theta}{dt} \right)^2 - l \sin \theta \frac{d^2 \theta}{dt^2} \right) \right) l \sin \theta \\ & - \left(m \left(\frac{d^2 x}{dt^2} - l \sin \theta \left(\frac{d\theta}{dt} \right)^2 + l \cos \theta \frac{d^2 \theta}{dt^2} \right) \right) l \cos \theta \\ & = I \frac{d^2 \theta}{dt^2}, \end{aligned} \quad (15)$$

po wymnożeniu nawiasów:

$$\begin{aligned} & \frac{mgl \sin \theta - ml^2 \sin \theta \cos \theta \left(\frac{d\theta}{dt} \right)^2}{- ml^2 \sin^2 \theta \frac{d^2 \theta}{dt^2} - ml \cos \theta \frac{d^2 x}{dt^2}} \\ & + \frac{ml^2 \cos \theta \sin \theta \left(\frac{d\theta}{dt} \right)^2 - ml^2 \cos^2 \theta \frac{d^2 \theta}{dt^2}}{ml \cos \theta \frac{d^2 x}{dt^2}} = I \frac{d^2 \theta}{dt^2}, \end{aligned} \quad (16)$$

podkreślone wyrazy można skrócić:

$$mgl \sin \theta - ml^2 \sin^2 \theta \frac{d^2 \theta}{dt^2} - ml \cos \theta \frac{d^2 x}{dt^2} - ml^2 \cos^2 \theta \frac{d^2 \theta}{dt^2} = I \frac{d^2 \theta}{dt^2}, \quad (17)$$

następnie wyłączyć przed nawias $ml^2 \frac{d^2 \theta}{dt^2}$:

$$mgl \sin \theta - ml^2 (\sin^2 \theta + \cos^2 \theta) \frac{d^2 \theta}{dt^2} - ml \cos \theta \frac{d^2 x}{dt^2} = I \frac{d^2 \theta}{dt^2} \quad (18)$$

wiadomo, że zawsze zachodzi:

$$\cos^2 \theta + \sin^2 \theta = 1, \quad (19)$$

więc można równanie uprościć do następującej formy:

$$\begin{aligned} mgl\sin\theta - ml^2 \frac{d^2\theta}{dt^2} - ml\cos\theta \frac{d^2x}{dt^2} &= I \frac{d^2\theta}{dt^2} \\ mgl\sin\theta - ml\cos\theta \frac{d^2x}{dt^2} &= (I + ml^2) \frac{d^2\theta}{dt^2}. \end{aligned} \quad (20)$$

W ten sposób otrzymano równania ruchu:

$$\boxed{\begin{cases} (M + m) \frac{d^2x}{dt^2} + \gamma_2 \frac{dx}{dt} = F + ml\sin\theta \left(\frac{d\theta}{dt}\right)^2 - ml\cos\theta \frac{d^2\theta}{dt^2} \\ (I + ml^2) \frac{d^2\theta}{dt^2} = mgl\sin\theta - ml\cos\theta \frac{d^2x}{dt^2} \end{cases}} \quad (21)$$

3.2.5 Linearyzacja w punkcie pracy

Zakładając sterowanie wahadłem w okolicy punktu pracy, gdzie wahadło będzie skierowane pionowo w górę na środku toru, czyli $x \approx 0$, $\theta \approx 0$, można dokonać przybliżenia:

$$\begin{aligned} \sin\theta &\approx \theta \\ \cos\theta &\approx 1 \\ \left(\frac{d\theta}{dt}\right)^2 &\approx 0. \end{aligned} \quad (22)$$

Dokonując takiego przybliżenia, można przepisać równania (21) ruchu następująco:

$$\begin{cases} (M + m) \frac{d^2x}{dt^2} + \gamma_2 \frac{dx}{dt} = F - ml \frac{d^2\theta}{dt^2} \\ (I + ml^2) \frac{d^2\theta}{dt^2} = mgl\theta - ml \frac{d^2x}{dt^2} \end{cases}. \quad (23)$$

Zakładając, że środek masy wahadła jest równy jego środkowi ciężkości, to $I = 0$ i dzieląc drugie równanie (23) przez ml , otrzymujemy:

$$\begin{cases} (M + m) \frac{d^2 x}{dt^2} + \gamma_2 \frac{dx}{dt} = F - ml \frac{d^2 \theta}{dt^2} \\ l \frac{d^2 \theta}{dt^2} = g\theta - \frac{d^2 x}{dt^2} \end{cases} \quad (24)$$

W celu przekształcenia równań tak, żeby zawierały tylko pochodne funkcji θ lub x , należy do pierwszego równania (24) podstawić obliczoną z drugiego wartość $\frac{d^2 \theta}{dt^2}$, czyli:

$$\frac{d^2 \theta}{dt^2} = \frac{g\theta}{l} - \frac{1}{l} \frac{d^2 x}{dt^2}. \quad (25)$$

W ten sposób otrzymujemy:

$$\begin{aligned} (M + m) \frac{d^2 x}{dt^2} + ml \left(\frac{g\theta}{l} - \frac{1}{l} \frac{d^2 x}{dt^2} \right) + \gamma_2 \frac{dx}{dt} &= F \\ (M + m) \frac{d^2 x}{dt^2} + mg\theta - m \frac{d^2 x}{dt^2} + \gamma_2 \frac{dx}{dt} &= F \\ M \frac{d^2 x}{dt^2} + \underline{m \frac{d^2 x}{dt^2}} + mg\theta - \underline{m \frac{d^2 x}{dt^2}} + \gamma_2 \frac{dx}{dt} &= F \\ M \frac{d^2 x}{dt^2} + \gamma_2 \frac{dx}{dt} &= F - mg\theta \quad /M \\ \frac{d^2 x}{dt^2} &= -\frac{mg}{M} \theta - \frac{\gamma_2}{M} \frac{dx}{dt} + \frac{1}{M} F. \end{aligned} \quad (26)$$

Podstawiając w ten sposób obliczoną pochodną do drugiego równania (24), otrzymujemy:

$$\begin{aligned} l \frac{d^2 \theta}{dt^2} &= g\theta + \frac{mg}{M} \theta + \frac{\gamma_2}{M} \frac{dx}{dt} - \frac{1}{M} F \quad /l \\ \frac{d^2 \theta}{dt^2} &= \frac{M + m}{Ml} g\theta + \frac{\gamma_2}{Ml} \frac{dx}{dt} - \frac{1}{Ml} F. \end{aligned} \quad (27)$$

Ostatecznie otrzymujemy równania:

$$\left\{ \begin{array}{l} \frac{d^2x}{dt^2} = -\frac{mg}{M}\theta - \frac{\gamma_2}{M}\frac{dx}{dt} + \frac{1}{M}F \\ \frac{d^2\theta}{dt^2} = \frac{M+m}{Ml}g\theta + \frac{\gamma_2}{Ml}\frac{dx}{dt} - \frac{1}{Ml}F \end{array} \right., \quad (28)$$

albo

$$\left\{ \begin{array}{l} \ddot{x} = -\frac{mg}{M}\theta - \frac{\gamma_2}{M}\dot{x} + \frac{1}{M}F \\ \ddot{\theta} = \frac{M+m}{Ml}g\theta + \frac{\gamma_2}{Ml}\dot{x} - \frac{1}{Ml}F \end{array} \right. \quad (29)$$

3.2.6 Reprezentacja macierzowa

Zakładając wektor stanu \vec{x} i wektor wejść \vec{u} następująco:

$$\vec{x} = \begin{bmatrix} x \\ \theta \\ \frac{dx}{dt} \\ \frac{d\theta}{dt} \end{bmatrix} = \begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix}, \quad \vec{u} = F, \quad (30)$$

Równanie macierzowe $\frac{d\vec{x}}{dt} = A\vec{x} + B\vec{u}$ można zapisać w sposób następujący:

$$\begin{bmatrix} \dot{x} \\ \dot{\theta} \\ \ddot{x} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{mg}{M} & -\frac{\gamma_2}{M} & 0 \\ 0 & \frac{M+m}{Ml}g & \frac{\gamma_2}{Ml} & 0 \end{bmatrix} \begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{M} \\ -\frac{1}{Ml} \end{bmatrix} F \quad (31)$$

4 Identyfikacja parametrów modelu

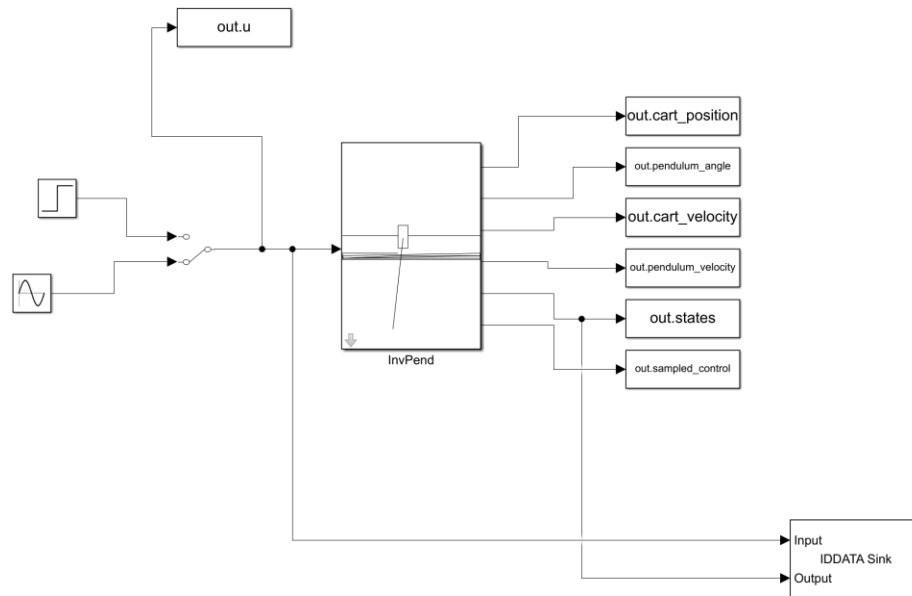
4.1 Pozyskanie danych do identyfikacji z rzeczywistego modelu

Ze względu na brak możliwości fizycznego dostępu do stanowiska wahadła odwróconego konieczne było skorzystanie z modelu wahadła rzeczywistego stworzonego w Simulinku (udostępnionego przez producenta):

```

T = 10; % czas symulacji
sim_step = 0.01; %Krok symulacji
out = sim('pendulum_dynamics.slx', T)

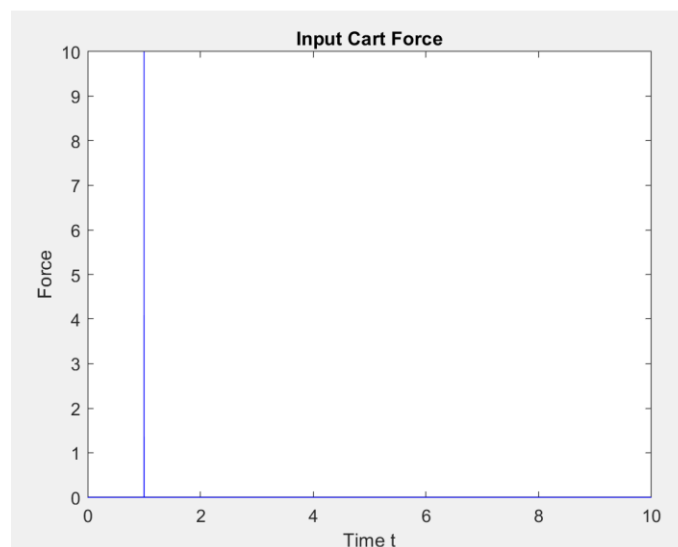
```



Rys. 3 Układ pomiarowy „rzeczywistego” obiektu

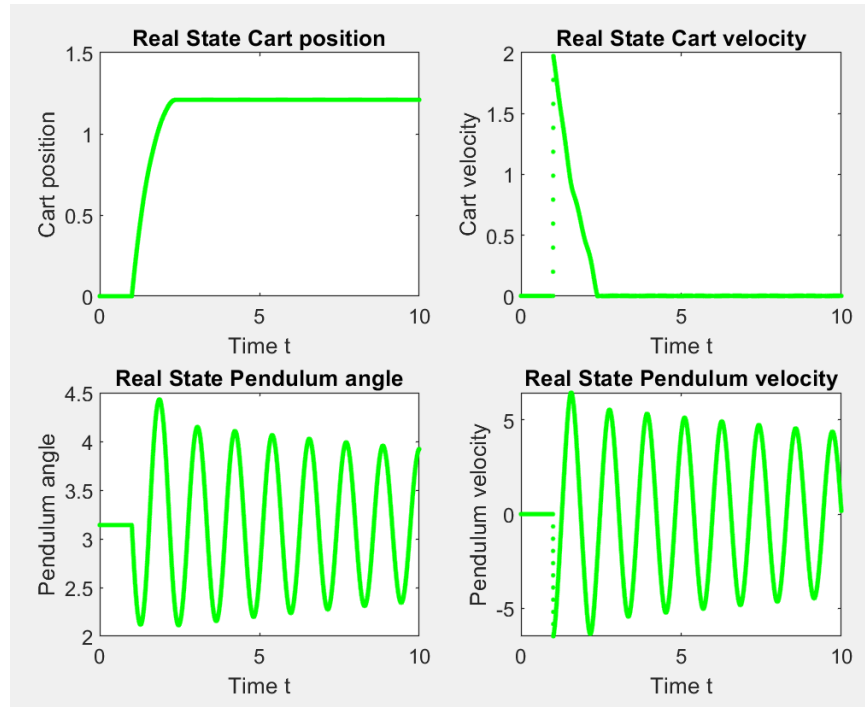
Blok IDDATA Sink pozwoli na zapisanie danych w odpowiedniej formie, co pozwoli je wykorzystać przez funkcję do identyfikacji nieznanymi parametrów modelu wahadła wyprowadzonego w poprzednim rozdziale.

Sygnał wymuszający podawany jako wejście do systemu:



Rys. 4 Sygnał wymuszający

Odpowiedź systemu na zadane wymuszenie:



Rys. 5 Odpowiedź systemu rzeczywistego na zadane wymuszenie

Dane zostały zarejestrowane i w kolejnych krokach zostaną wykorzystane w procesie identyfikacji.

4.2 Stworzenie obiektu wahadła w Matlab

Proces identyfikacji został przeprowadzony na modelu nieliniowym wahadła (21), wyprowadzonym w pierwszej części projektu, ponieważ estymacja parametrów modelu musiała zostać przeprowadzona w pełnym zakresie pracy wahadła (model liniowy jest stabilny tylko w obszarze punktu pracy). Wahadło ma formę jednorodnego pręta, o momencie bezwładności:

$$I = \frac{1}{3} * m * l^2 \quad (32)$$

Po podstawieniu do układu równań (21), model nieliniowy przyjmuje postać:

$$\begin{cases} \frac{d^2x}{dt^2} = \frac{1}{M+m} * \left(F + m * l * \sin\theta * \left(\frac{d\theta}{dt} \right)^2 - m * l * \cos\theta * \frac{d^2\theta}{dt^2} - \mu \frac{dx}{dt} \right) \\ \frac{d^2\theta}{dt^2} = \frac{2}{3 * l^2} * (g * l * \sin\theta - l * \cos\theta * \frac{d^2x}{dt^2}) \end{cases} \quad (33)$$

Algorytm estymujący wymagał przedstawienia struktury modelu w specjalnej formie i stworzenia funkcji w języku MATLAB, która jako argumenty przyjmuje między innymi parametry, które są przedmiotem estymacji oraz sygnał wejściowy:

```

function [dx, y] = non_lin_inverted_pendulum(t, x, u, l, fi, varargin)
m = 0.12;
M = 0.5723;
% M = 0.7723;
g = 9.81;

% x(2) = x(2) * (-1) + pi;
% x(2) = x(2) * (-1);
% x(4) = x(4) * (-1);

% Output equations.
y = [x(1); ... % Cart position.
    ((-1) * x(2)) + (2*pi); ... % Pendulum angle.
    x(3); ... % Cart velocity.
    (-1) * x(4) ... % Pendulum angular velocity.
    ];

% State equations.
dx = [x(3); ... % Cart velocity.
    x(4); ... % Pendulum angular velocity.
    ( ( 4*sin(x(2))*l*m*(x(4)^2) ) + ( 4*u ) - ( 4*fi*x(3) ) - (
3*g*sin(x(2))*cos(x(2))*m ) ) / ( ( -3*m*(cos(x(2))^2 ) + ( 4*M ) + ( 4*m ) );
    ( 3 * ( ( -sin(x(2))*cos(x(2))*l*m*(x(4)^2) ) - ( u*cos(x(2)) ) + (
M*g*sin(x(2)) ) + ( g*sin(x(2))*m ) + ( fi*cos(x(2))*x(3) ) ) ) / ( 1 * ( ( -
3*m*(cos(x(2))^2 ) + ( 4*M ) + ( 4*m ) ) )
    ];
end

```

4.3 Identyfikacja parametrów modelu nieliniowego

Proces identyfikacji parametrów modelu przebiegał w wielu etapach, a jego precyzyjność zapewnia wykorzystanie toolbox'a *System Identification* z pakietu Matlab. W pierwszej kolejności należy na podstawie utworzonego wcześniej modelu, stworzyć specjalny obiekt nieliniowy *nlgr*, który zostanie podany jako argument do funkcji estymującej:

```

file_name = 'non_lin_inverted_pendulum';
Order = [4 1 4];
% Parameters = [1; 0.41]; % l fi
% Parameters = [0.3; 0.5]; % l fi
% Parameters = [0.2; 0.5]; % l fi

Parameters = [0.3; 0.6]; % l fi
InitialStates = [0; pi; 0; 0]; % x theta dx/dt dtheta/dt
nlgr = idnlgrey(file_name, Order, Parameters, InitialStates, 0);
nlgr.SimulationOptions.AbsTol = 1e-12;
nlgr.SimulationOptions.RelTol = 1e-15;

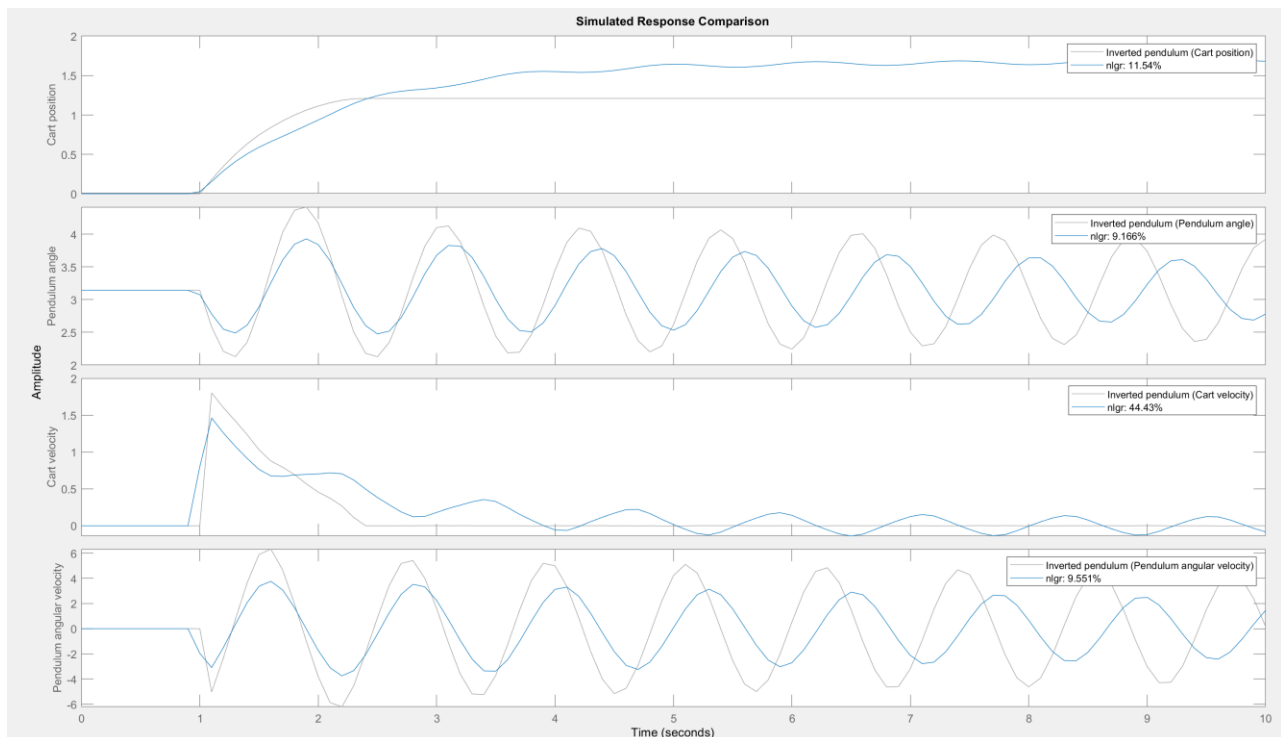
```

Dodatkowo podaje się również wartości początkowe szukanych parametrów, warunki początkowe oraz opcje estymacji. Kolejnym krokiem jest porównanie dokładności nieliniowego modelu wyprowadzonego poprzednio do symulowanego modelu rzeczywistego:

```
figure('Name', [z.Name ': Inverted pendulum model before estimation']);

compare(z, nlgr);
```

Przedstawienie dokładności przed estymacją:



Rys. 6 Dokładność modelu przed estymacją

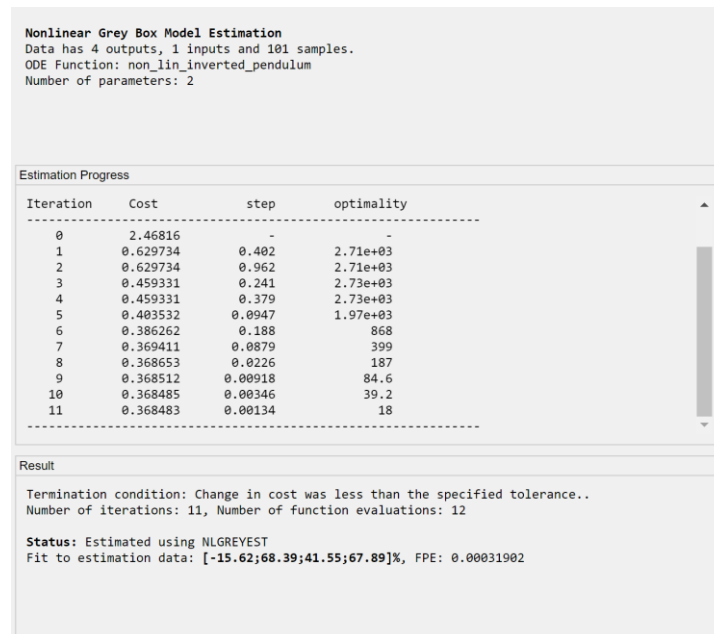
Następnie przechodzimy do wykonania estymacji:

```
opt = nlgreyestOptions('Display', 'on', 'SearchMethod', 'auto');

opt.SearchOptions.MaxIterations = 1000;

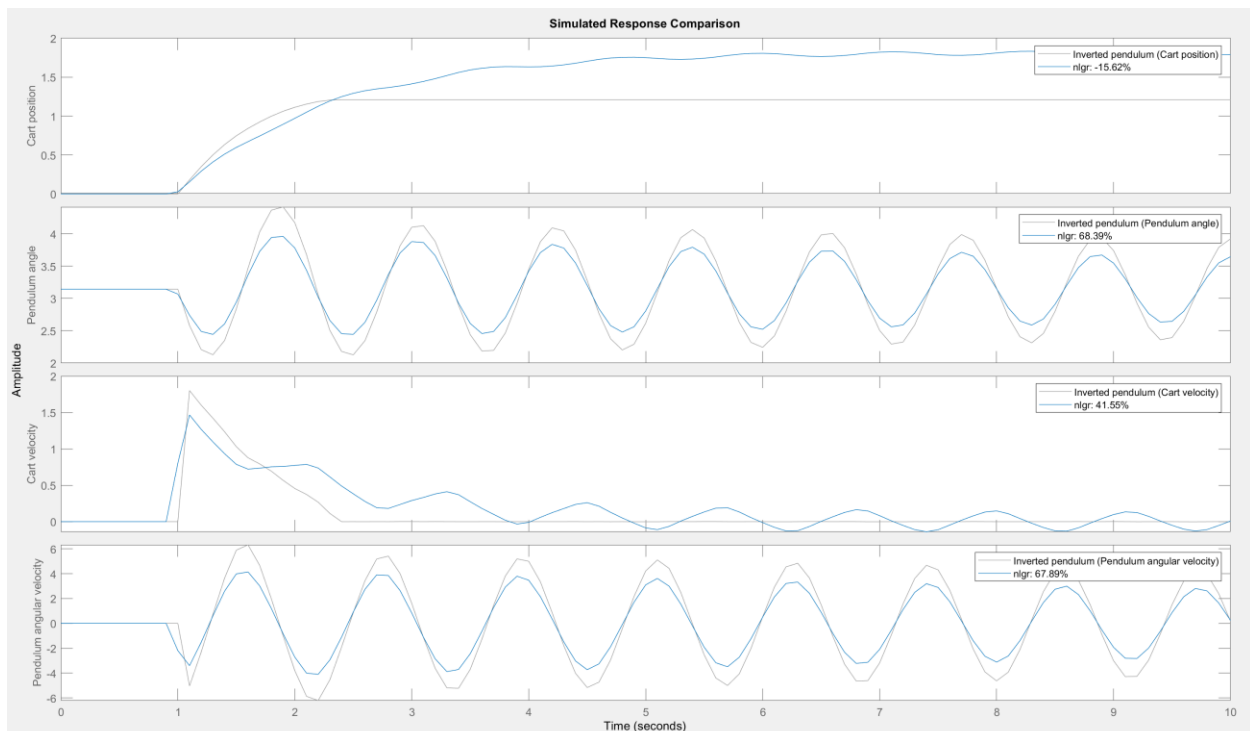
nlgr = nlgreyest(z, nlgr, opt);
```

Rejestracja przebiegu estymacji:



Rys. 7 Rejestracja przebiegu estymacji

Prezentacja dokładności modelu po estymacji wartości szukanych parametrów:



Rys. 8 Wynik estymacji parametrów modelu

Proces estymacji znacząco poprawił jakość odpowiedzi wahadła, natomiast dokładność pozycji wózka została nieznacznie pogorszona. Prawdopodobną przyczyną jest zbyt niska dokładność modelu.

Otrzymany wynik uznano jednak za wystarczająco dobrze dopasowany. Wartości szukanych parametrów w tym przypadku wyniosły: współczynnik tarcia - 0.55, długość wahadła - 0.26.

5 Algorytm sterujący

5.1 Cele algorytmu i założenia

Algorytm sterujący ma kilka zadań. Najważniejszym jest wychylenia wahadła ze stabilnego punktu równowagi „pionowo w dół” ($\theta = \pi$) i stabilizacja w punkcie niestabilnym, czyli pionowo w górę. Dodatkowo algorytm powinien ustawiać wahadło na środku toru ($x = 0$) oraz zapobiegać przekroczeniu ograniczeń mechanicznych, czyli długości toru. Ponadto w projekcie została podjęta próba wyznaczenia i zamodelowania sterowania (stabilizacji) optymalnego na podstawie zidentyfikowanego modelu liniowego i rozwiązania równania Riccatiego.

5.2 Normalizacja kąta wychylenia wahadła

Jednym z pierwszych kroków podjętych podczas tworzenia algorytmu sterowania było utworzenie funkcji normalizującej kąt wychylenia wahadła. Jej celem jest przekształcenie kąta wychylenia wahadła do zakresu $(-\pi, \pi)$, dzięki czemu możliwa jest stabilizacja w kątach będących wielokrotnościami liczby 2π . Na poniższym listingu został zamieszczony fragment funkcji konwertującej kąt.

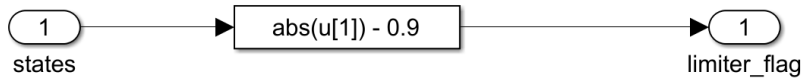
```
function angle = angle_convert(angle_raw)
    if angle_raw < 0
        is_negative = 1;
    else
        is_negative = 0;
    end
    if ~is_negative
        q = fix(angle_raw/2*pi);
        rem = mod(angle_raw,2*pi);
        if (rem >= pi && rem <= 2*pi)
            angle = -(2*pi - rem);
        else
            if q > 0
                angle = rem;
            else
                angle = angle_raw;
            end
        end
    end
```

Funkcja działa w taki sposób, że w przypadku dodatnich kątów zamienia wartości z 2 i 3 ćwiartki na zakres $(0, \pi)$, natomiast z ćwiartki 1 i 4 na zakres $(-\pi, 0)$. Wykorzystywane w tym celu jest wyznaczanie ilorazu oraz reszty z dzielenia przez 2π . Dla liczb ujemnych przekształcanie zrealizowane jest w sposób analogiczny.

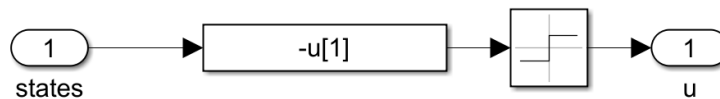
5.3 Ograniczenie położenia wózka

W rzeczywistym układzie wahadła na wózku tor, po którym porusza się wózek ma ograniczoną długość. Z tego powodu należy zadbać o to, aby wartość położenia x nie przekraczała granicznych wartości.

Rozważany model ma długość toru $R = 1.8m$, dlatego w przypadku osiągnięcia końca toru, czyli kiedy $x = \pm 0.9m$ lub trochę mniej, zamiast obliczanego na podstawie algorytmu sterowania, podawane jest $u = \pm 1$ w celu „odsunięcia się” od końca toru. Sposób implementacji przedstawiony jest poniżej.



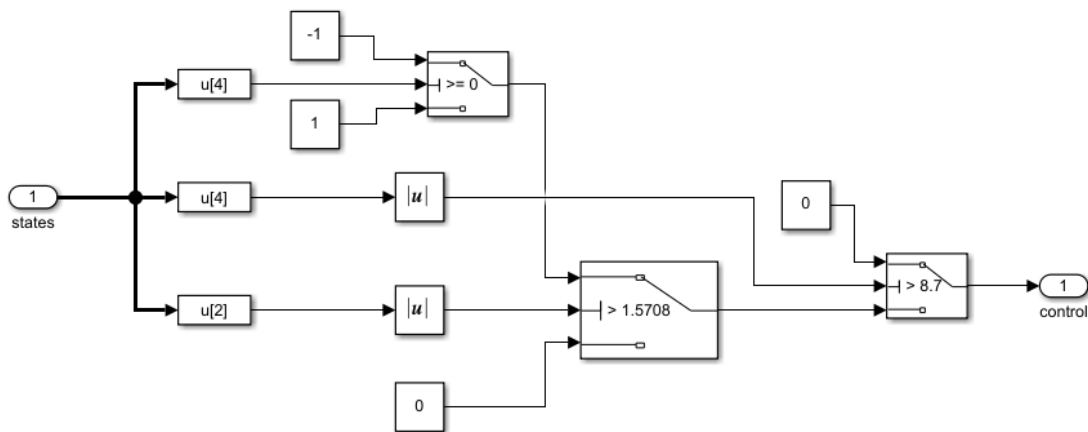
Rys. 9 Sprawdzenie zakresu pierwszej zmiennej stanu (x – położenie wózka na torze)



Rys. 10 Sterowanie w przeciwną stronę do wykrytego ograniczenia

5.4 Rozbujanie wahadła

Do głównych zadań kontrolera oprócz stabilizacji w punkcie równowagi należy rozbujanie wahadła, aby znalazło się w okolicy punktu pracy, czyli w takim otoczeniu punktu pracy ($x = 0, \theta = 0, \dot{x} = 0, \dot{\theta} = 0$), dla którego opracowany na podstawie przybliżenia modelu nieliniowego do liniowego algorytm stabilizacji będzie skuteczny. Zaimplementowany algorytm został nazwany „swinging controller” i zaprezentowany poniżej.



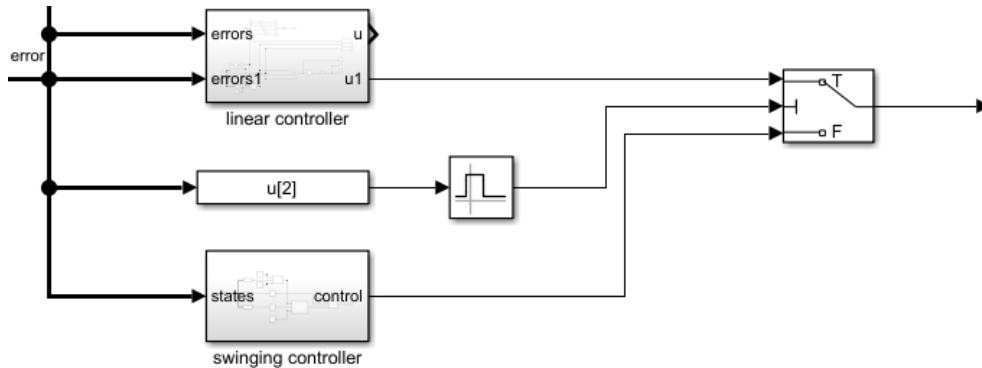
Rys. 11 „swinging controller” – rozbujanie wahadła

Zasada działania jest następująca: kiedy prędkość kątowa wahadła jest większa lub równa 0, to podawane jest sterowanie $u = -1$, dzięki czemu do wahadła jest dostarczana dodatkowa energia kinetyczna. Kiedy zostanie osiągnięty punkt krańcowy wychylenia i zaczyna opadać, a co za tym idzie prędkość kątowa zmienia znak, to podawane jest sterowanie o przeciwnym znaku. W ten sposób za każdym razem wahadło wychyla się dalej niż poprzednio, aż do osiągnięcia strefy, w której zastosowanie ma regulator

stabilizujący. Ponadto zostało dodane ograniczenie sterowania, gdy wahadło przekroczy kąt $\pm \frac{\pi}{2}$ oraz gdy prędkość kątowa przekroczy pewną wartość w celu uniknięcia sytuacji, w której wahadło będzie miało zbyt dużą energię kinetyczną, aby dało się je wyhamować regulatorem stabilizującym.

5.5 Przełączanie kontrolerów stabilizującego i wychylającego

Wybór aktywnego kontrolera, na podstawie którego wyznaczane jest sterowanie odbywa się poprzez sprawdzanie aktualnego kąta wychylenia wahadła. Kiedy kąt mieści się w zakresie stabilizacji, to działa regulator stabilizujący, w przeciwnym przypadku – wychylający. Przełączanie zostało zrealizowane z użyciem bloku „interval test” oraz „switch”, jak zostało pokazane na poniższym rysunku.



Rys. 12 Przełączanie kontrolerów

5.6 Regulator stabilizujący

Najważniejszym elementem sterownika jest regulator stabilizujący, który ma za zadanie doprowadzenie wahadła do niestabilnego punktu równowagi i utrzymanie go tam, czyli kompensacja zakłóceń. W projekcie została podjęta próba wyznaczenia regulatora optymalnego z równań Riccatiego na podstawie zlinearyzowanego modelu oraz zidentyfikowanych parametrów. Zastosowany został także regulator PID wyznaczający sterowanie na podstawie uchybów wartości stanów od wartości zadanej.

5.6.1 Regulator LQ

Dzięki przeprowadzonej identyfikacji nieznanych parametrów modelu uzyskano dokładne macierze stanu, na podstawie których możliwe było wyznaczenie wzmacnień regulatora LQ. Na poniższym listingu został zamieszczony sposób wyznaczania tych wzmacnień.

```
% state matrices after linearization
A = [0      0      1      0;
     0      0      0      1;
     0      -(m*g/MM)  -gamma2/MM  0;
     0      g*((MM+m)/(MM*1)  gamma2/(MM*1)  0];
B = [0 0 1/MM -1/(MM*1)]';
C = [1 1 1 1];
D = 0;

% observability and controllability test
ob = obsv(A,C);
rank(ob); % n=4: observable
```

```

ct = ctrb(A,B);
rank(ct);                                % n=4: controllable

% lqr coefficients calculation
Q = [1 0 0 0;
     0 100 0 0;
     0 0 10 0;
     0 0 0 100];
R = 1;

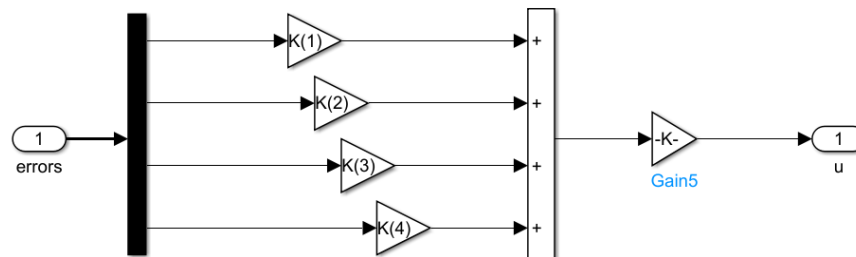
K = lqr(A,B,Q,R);                        % LQ regulator parameters

```

Zmienne widoczne na listingu mają następujące znaczenie: m oraz MM oznaczają odpowiednio masę wahadła oraz wózka, g jest przyspieszeniem ziemskim, l oraz γ są identyfikowanymi w poprzednim rozdziale parametrami, A , B , C , D są macierzami stanu, natomiast Q , R i K odpowiednio macierzą wag, ograniczeniem sterowania oraz obliczanym wektorem wzmocnień dla regulatora. W ten sposób uzyskano następujące parametry regulatora proporcjonalnego:

```
K = [-1, -41.2832, -4.4032, -13.3673];
```

Mając wyznaczone wzmocnienia można je w prosty sposób zaimplementować w algorytmie sterującym, jak pokazano na poniższym rysunku.

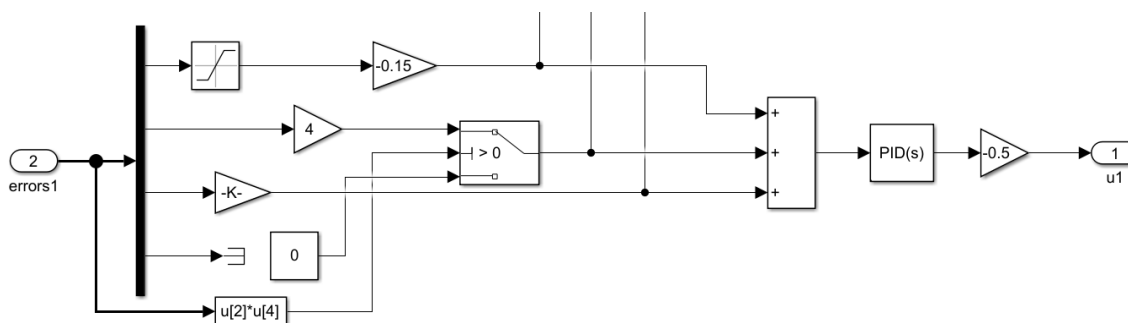


Rys. 13 Implementacja regulatora LQR

Niestety wyznaczony regulator LQ nie był wystarczająco dokładny i nie doprowadzał w skończonym czasie uchybów do zera. Prawdopodobną przyczyną jest niewystarczająco dokładne odwzorowanie rzeczywistego obiektu w sposób matematyczny, w szczególności niedokładnie dobrane parametry tego modelu podczas identyfikacji, na etapie której już były widoczne pewne rozbieżności.

5.6.2 Regulator PID

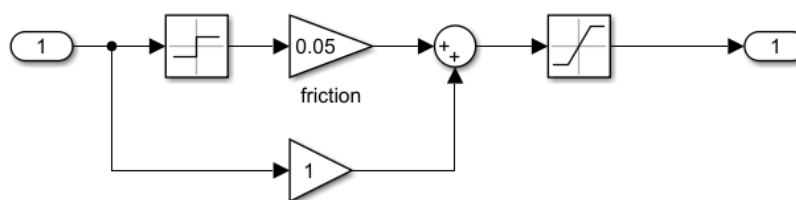
Oprócz regulatora LQ podjęto próbę zastosowania klasycznego regulatora PID, którego zadaniem jest sprowadzanie uchybu do zera. Na jego wejściu podawana jest kombinacja uchybów poszczególnych zmiennych stanu ze współczynnikami wyznaczonymi empirycznie. Ponadto wprowadzona została saturacja uchybu położenia, żeby nie „utrudniał” on stabilizacji wahadła w pionie oraz zerowanie uchybu kąta, jeśli ma on przeciwny znak względem prędkości, gdyż w takim przypadku kąt i tak zmierza do zera. Schemat układu z regulatorem PID został zamieszczony na poniższym rysunku.



Rys. 14 Stabilizacja z wykorzystaniem regulatora PID

5.7 Saturacja i kompensacja tarcia

Kompensacja tarcia oraz saturacja została zrealizowana poprzez nieznaczne zwiększenie sterowania o wartość proponowaną w instrukcji do stanowiska (0.05) oraz ograniczenie sterowania obliczanego wszystkimi algorytmami do zakresu $(-1,1)$.



Rys. 15 Kompensacja tarcia i saturacja sterowania

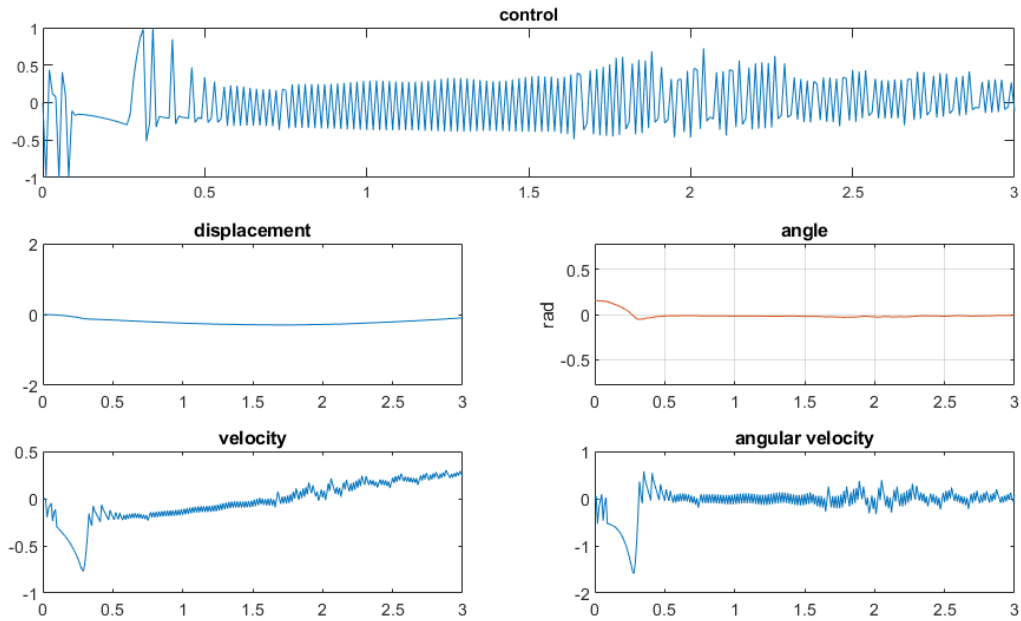
6 Symulacja stabilizacji wahadła

6.1 Stabilizacja w otoczeniu punktu pracy

W pierwszej symulacji sprawdzone zostało działanie algorytmu stabilizacji pozycji wahadła w punkcie pracy. Symulacja została przeprowadzona z warunkami początkowymi:

```
init_cond = [0 pi/20 0 0];
stab_point = [0 0 0 0];
```

Aktywnym regulatorem jest regulator PID. Na poniższym wykresie można zaobserwować stabilizację z horyzontem czasowym 3s.



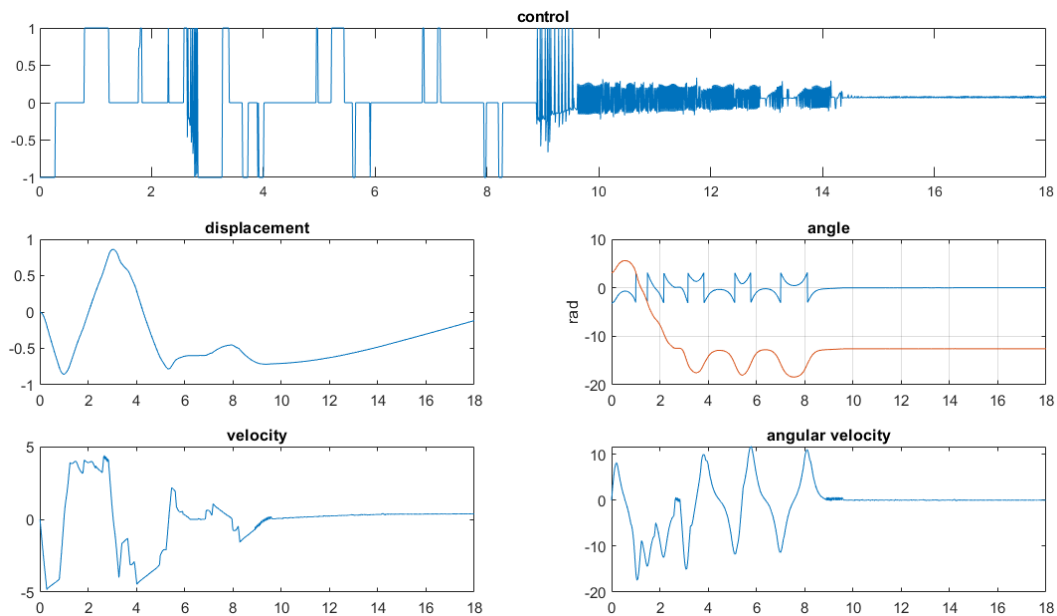
Rys. 16 Symulacja regulatora stabilizującego, warunek początkowy $\theta_0 = \frac{\pi}{20}$

6.2 Podniesienie wahadła i stabilizacja

Kolejna symulacja obejmuje rozbujaanie wahadła i stabilizację w punkcie pracy. Warunki początkowe są następujące:

```
init_cond = [0 pi 0 0];
stab_point = [0 0 0 0];
```

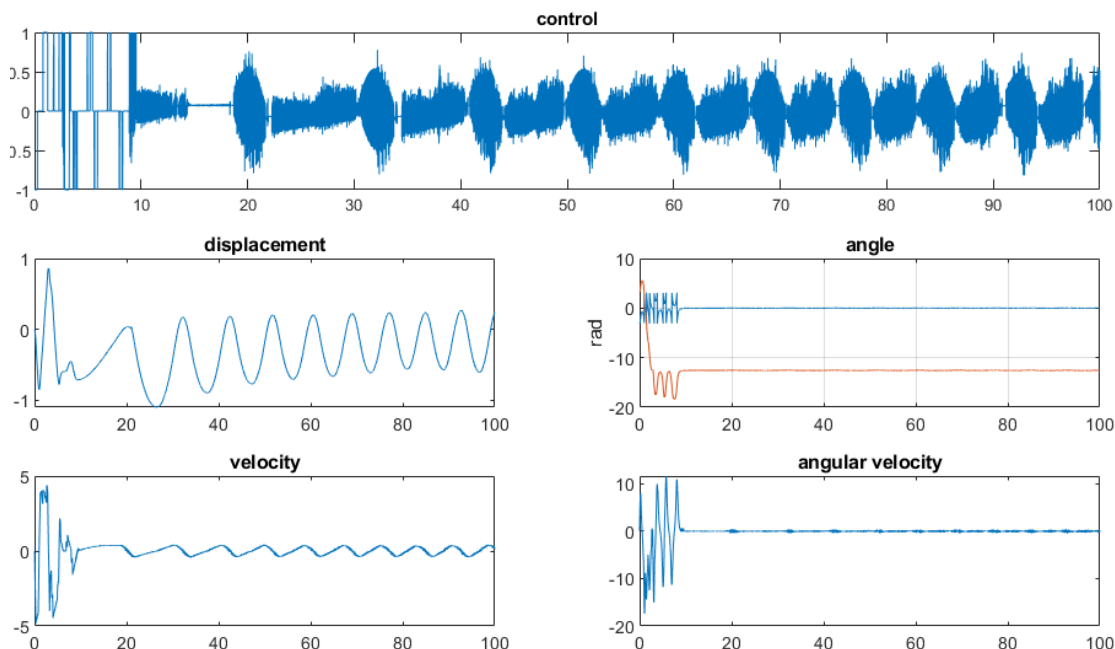
Symulacja została przeprowadzona z horyzontem czasowym 18s.



Rys. 17 Symulacja wychylenia i stabilizacji wahadła

Na powyższych wykresach można zaobserwować, że wszystkie zmienne stanu zostały sprowadzone do 0, jednak nie udało się tego osiągnąć przy pierwszej „okazji”, czyli za pierwszym razem, kiedy wahadło weszło w strefę stabilizacji. Udało się to osiągnąć dopiero wtedy, kiedy prędkość wózka i prędkość kątowa nie są zbyt duże w chwili osiągnięcia zakresu pracy regulatora PID.

W celu sprawdzenia, czy po pewnym czasie uchyby i sterowanie dążą do zera przeprowadzono symulację dla dłuższego czasu. Wynik został zamieszczony na poniższym wykresie.



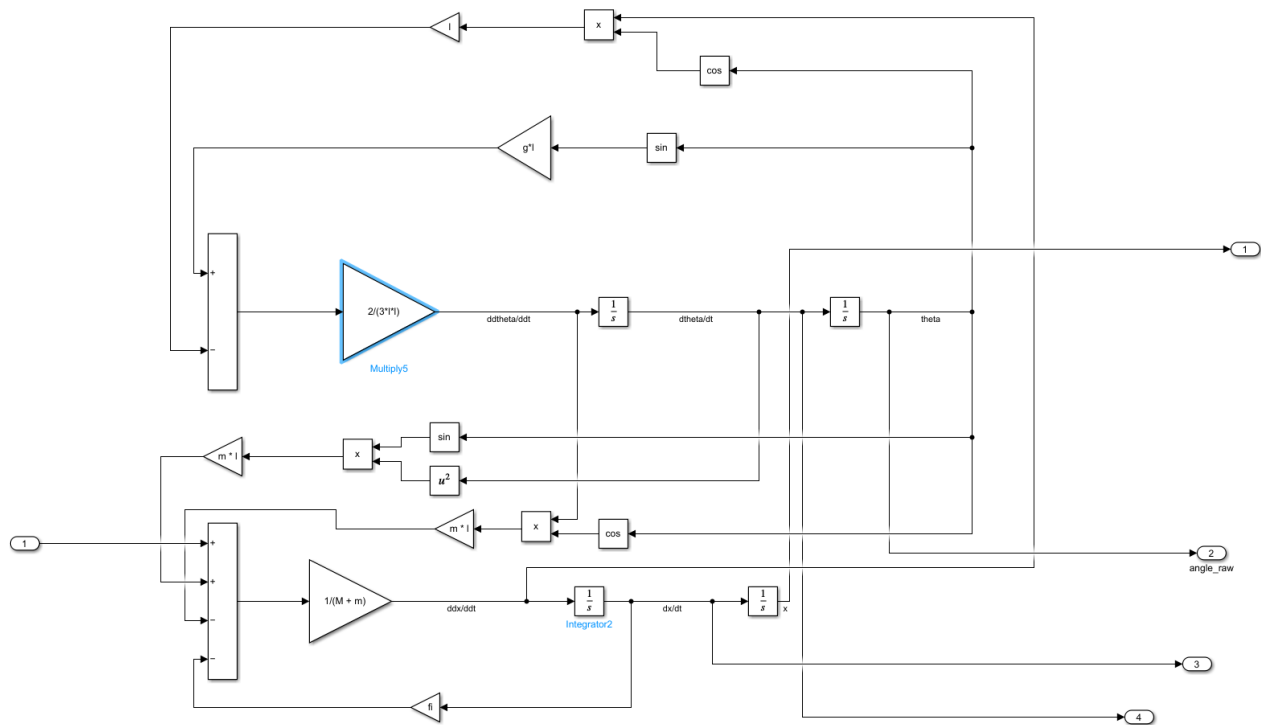
Rys. 18 Symulacja wychylenia i stabilizacji wahadła dla $t=100$

Tym razem można zaobserwować, że kąt i prędkość kątowa jest prawidłowo sprowadzana do 0, jednak wózek okresowo przemieszcza się w pewnym zakresie toru. Prawdopodobną przyczyną takiego zachowania jest niezgodność modelu matematycznego i obiektu, na którym były przeprowadzane testy.

W celu sprawdzenia tej hipotezy uruchomiono algorytm sterujący dla modelu nieliniowego danego równaniami (21).

6.3 Symulacja stabilizacji wahadła dla modelu nieliniowego

W trakcie testowania działania algorytmu sterowania przyjęto model nieliniowy (33) (tak samo jak podczas procesu identyfikacji parametrycznej modelu), że wahadło ma formę jednorodnego pręta. Następnie na tej podstawie został stworzony model nieliniowy wahadła w programie Simulink, który posłużył do testowania oraz dostrajania algorytmu sterowania:



Rys. 19 Model nieliniowy wahadła odwróconego

Parametry układu wyznaczone podczas identyfikacji oraz parametry symulacji modelu są pobierane z przestrzeni *workspace* z Matlab:

```
clear all;
close all;
clc;
beep off;

T = 500; % s
step = 0.01; % s
int_tolerance = 0.001;

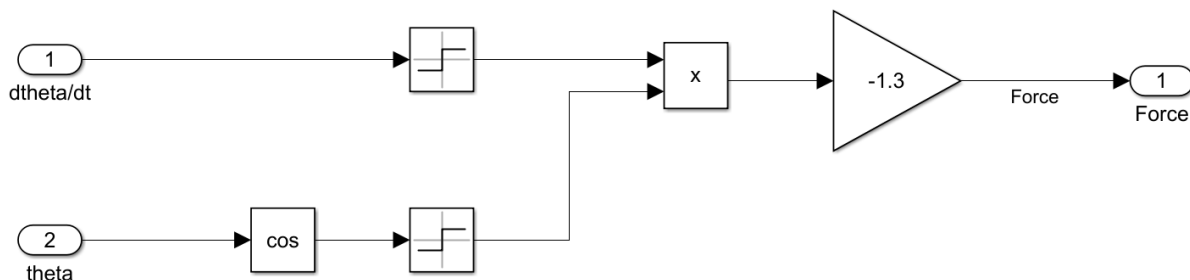
l = 0.512166;
fi = 0.267534;

g = 9.81;

M = 0.5723;
m = 0.12;

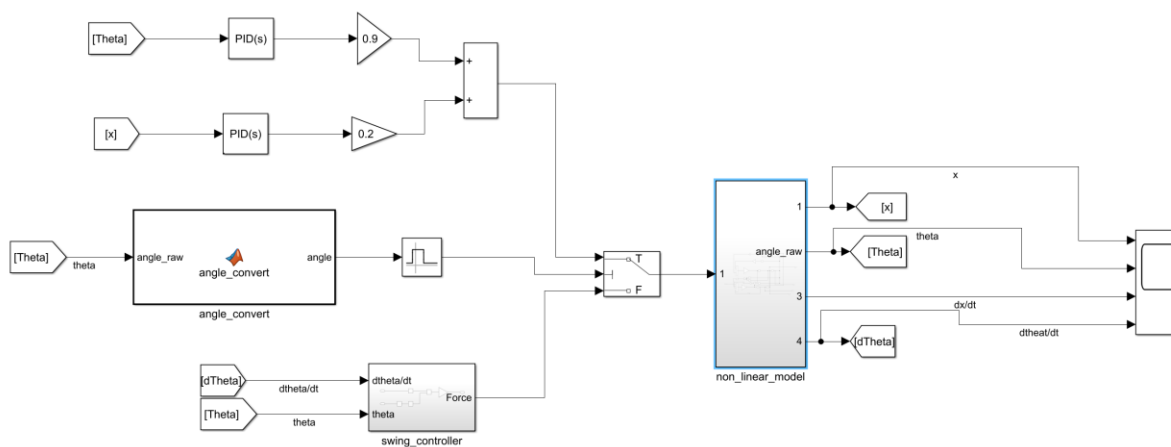
init_theta = pi;
```

Proces sterowania wahadłem odwróconym został podzielony na dwie niezależne części. W pierwszej kolejności, kiedy wahadło znajduje się w dole jest uruchamiany regulator *swing_controller*, którego zadaniem jest przemieszczenia wahadła do pozycji pionowej w jak najkrótszym czasie.



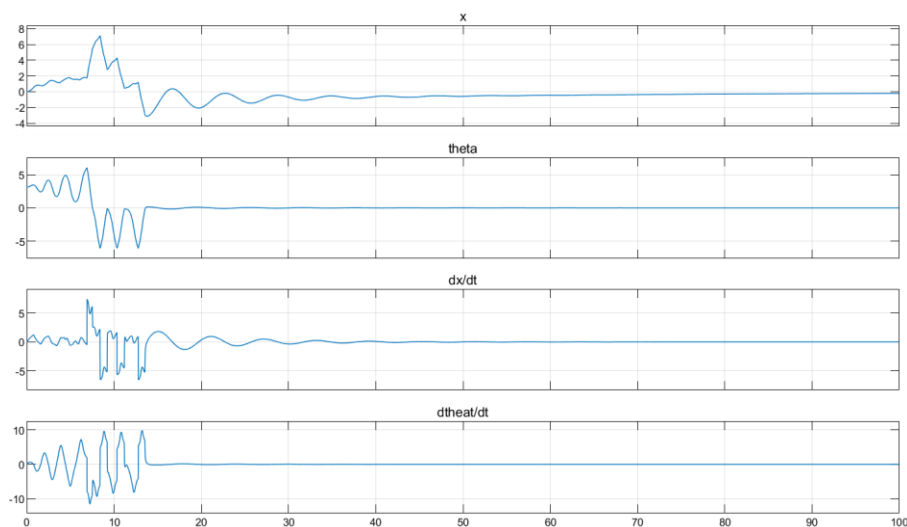
Rys. 20 Swing controller dla modelu nieliniowego

Gdy wahadło znajdzie się blisko punktu pracy, rozpoczyna się proces stabilizacji wahadła w pionie. Do tego zadania został wykorzystany regulator, złożony z połączonych równolegle regulatorów PID. Jako wejścia do regulatorów został podany sygnał położenia wózka oraz kąta wychylenia wahadła. Sygnały sterujące z każdego regulatora, następnie są mnożone przez współczynnik wagi danego sygnału (sterowanie wychyleniem wahadła jest o wiele bardziej znaczące niż położenie wózka, co oznacza, że w pierwszej kolejności regulator będzie do ustabilizowania pozycji wahadła, a następnie wózka). Oba sygnały sterujące są sumowane i podłączone jako wejście do systemu. Przetaczanie pomiędzy regulatorami jest zrealizowane w bardzo prosty sposób. Wahadło po wejściu w obszar pomiędzy $(-0.2\text{rad}, 0.2\text{rad})$ (punkt pracy wahadła) są uruchamiane regulatory w celu stabilizacji w punkcie pracy, natomiast poza tym obszarem uruchamiany jest drugi regulator, który stara się wprowadzić wahadło w ten obszar.



Rys. 21 System sterowania wahadłem odwróconym w postaci modelu nieliniowego

Wyniki działania stworzonego regulatora zostały zaprezentowane poniżej:



Rys. 22 Wyniki działania regulatora

Wózek w chwili $t = 0$ jest na środku (w pozycji $x = 0$), a wahadło w dole, czyli kąt obrotu wahadła wynosi $\theta = \pi$. W pierwszej kolejności, zgodnie z założeniem, zostaje uruchomiona część regulatora odpowiedzialna za umieszczenie wahadła w górze. Po wykonaniu kilku wahaniec oraz wykonaniu jednego pełnego obrotu wahadło wpada w obszar, w którym aktywna staje się część układu sterowania odpowiedzialna za utrzymanie wahadła w pionie. Stan ten pozostaje utrzymany już do końca symulacji (próby z wydłużeniem czasu symulacji dawały dokładnie ten sam efekt).

7 Wnioski

Opracowany algorytm sterowania działa poprawnie dla modelu matematycznego wyznaczonego w trzecim rozdziale. Próby zastosowania go do obiektu bardziej zbliżonego do rzeczywistego (który należało w projekcie potraktować jako rzeczywisty) nie przyniosły pozytywnych efektów i nie sprawdzał się. Z tego powodu należało wprowadzić dodatkowe mechanizmy, które w pewnym stopniu niwelowały różnice wynikające z niedokładności modelu. Ponadto konieczne było zastosowanie pewnych ograniczeń, które miały zapobiegać uszkodzeniom mechanicznym wahadła w rzeczywistym układzie pracy (uderzenie w granicę toru, rozpędzenie do bardzo dużych prędkości kątowych) oraz saturacji sterowania.

Ostatecznie dla rzeczywistego obiektu udało się częściowo zrealizować postawione zadanie. Kąt wahadła prawidłowo jest sprowadzany do 0, natomiast pozycja wózka wchodzi w oscylacje w obrębie toru. Próby dopracowania nastaw regulatora i wzmocnień nie przyniosły jednak poprawy, a zaprezentowany w poprzednim rozdziale rezultat jest najlepszym osiągniętym podczas pracy nad wahadłem.

Przedstawione wcześniej nieprawidłowości (oscylacje wózka, nieudana stabilizacja przy pierwszym osiągnięciu strefy liniowej) wynikają prawdopodobnie z nagromadzenia niedokładności otrzymywanych na kolejnych etapach projektu. Na etapie wyznaczania modelu matematycznego zostały dokonane pewne przybliżenia i uproszczenia, następnie podczas identyfikacji również nie zostało uzyskane dokładne przybliżenie, jedynie akceptowalne. Natomiast na etapie implementacji sterowania należało w sposób empiryczny dobrać nastawy regulatora oraz kluczowe wzmocnienia. Wszystko to doprowadziło do tego, że zaprojektowany sterownik nie spełnia wszystkich postawionych założeń i jest daleki od optymalności.