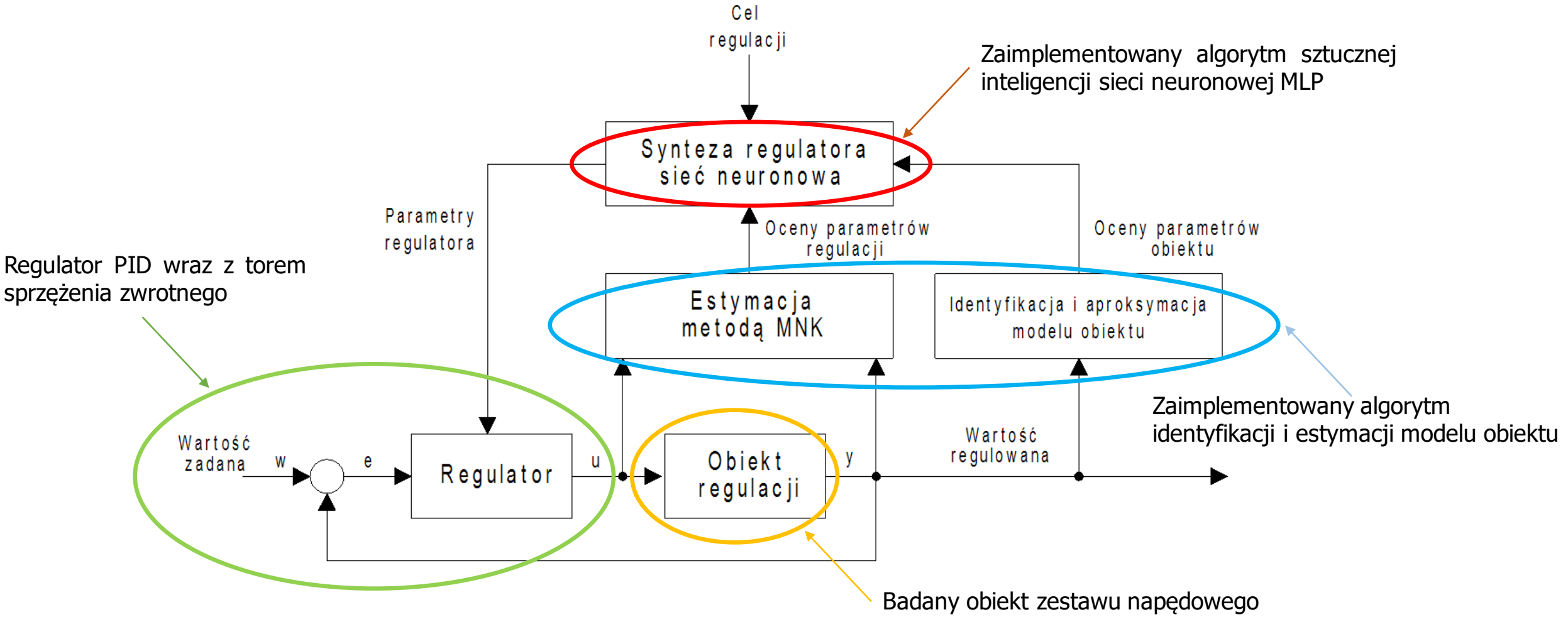


System regulacji adaptacyjnej z identyfikacją napędu elektrycznego

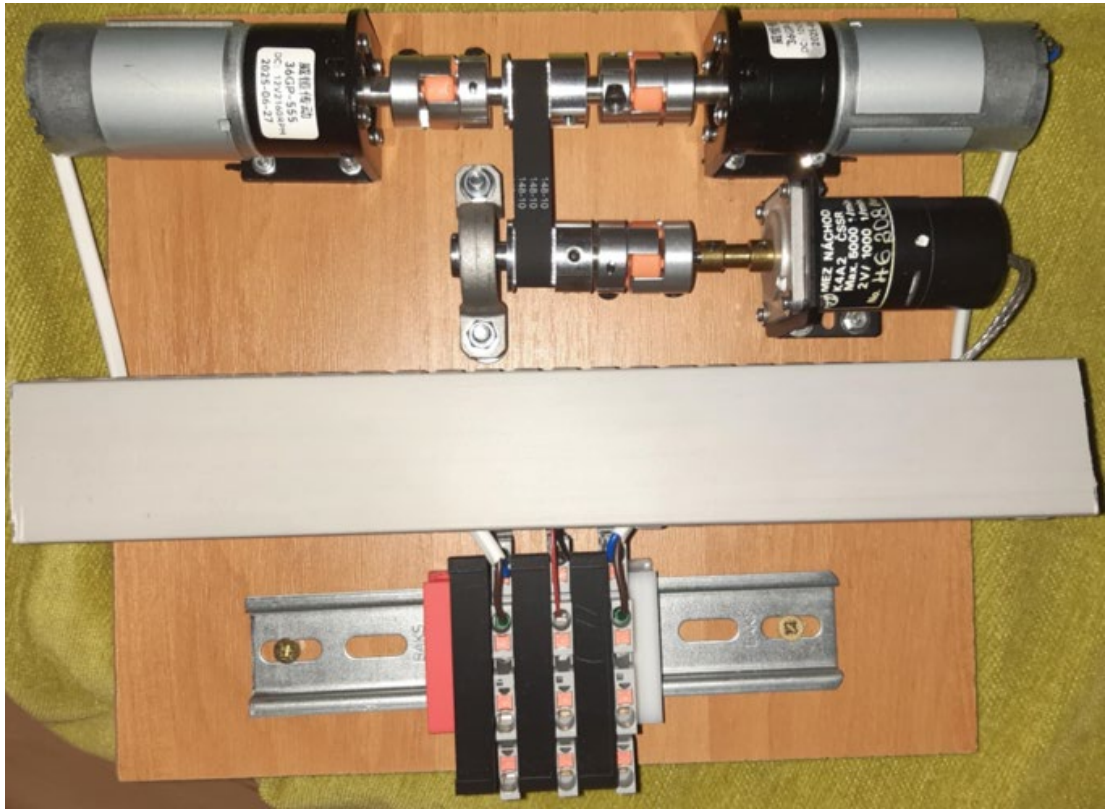
Autor: mgr inż. Filip Mazurek

Promotor: dr inż. Leszek Kasprzyczak

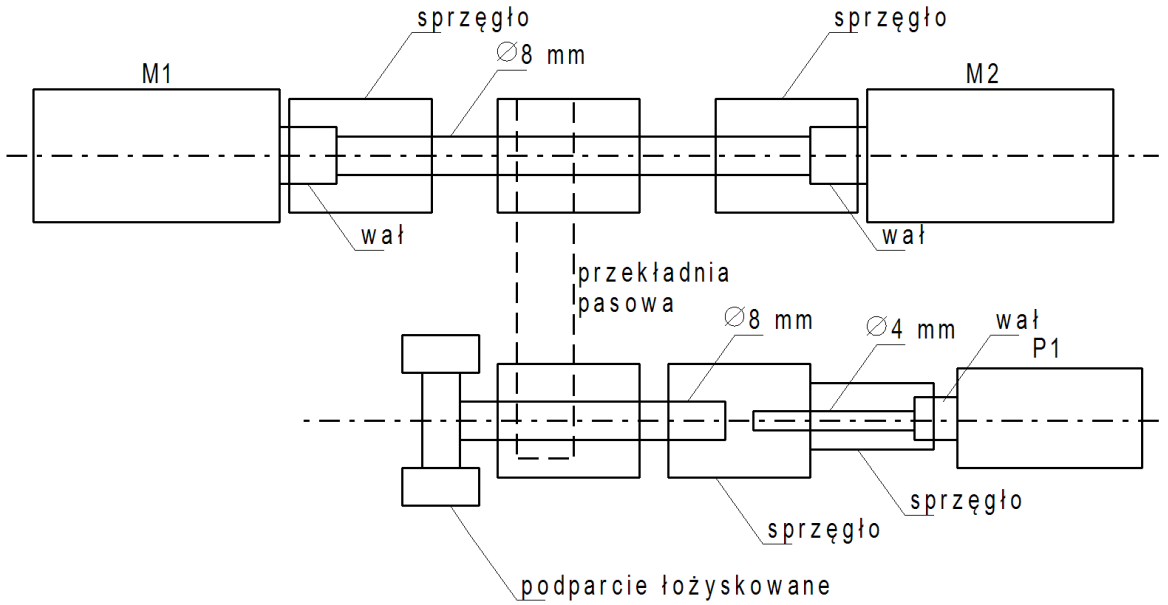
Politechnika Śląska



Rys. Schemat regulacji z wykorzystaniem sieci neuronowej

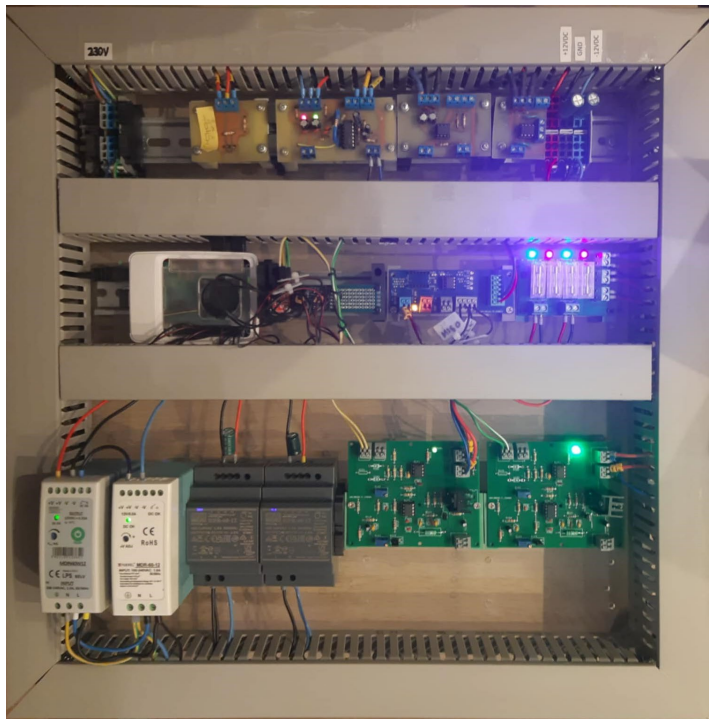


Rys. Widok zmontowanych silników i tachogeneratora

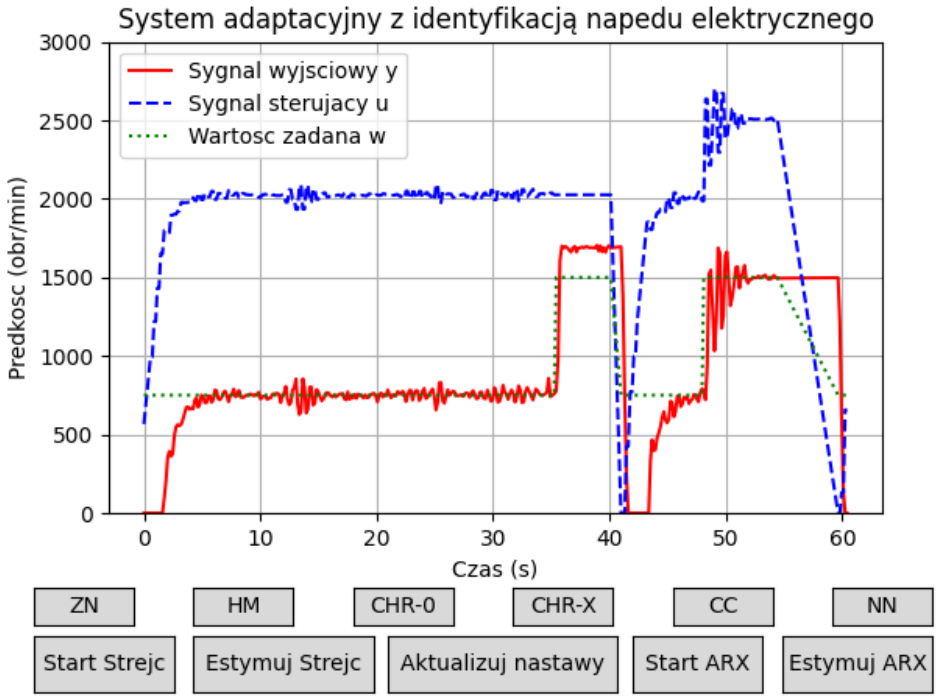


Rys. Schemat mechaniczny zestawu napędowego

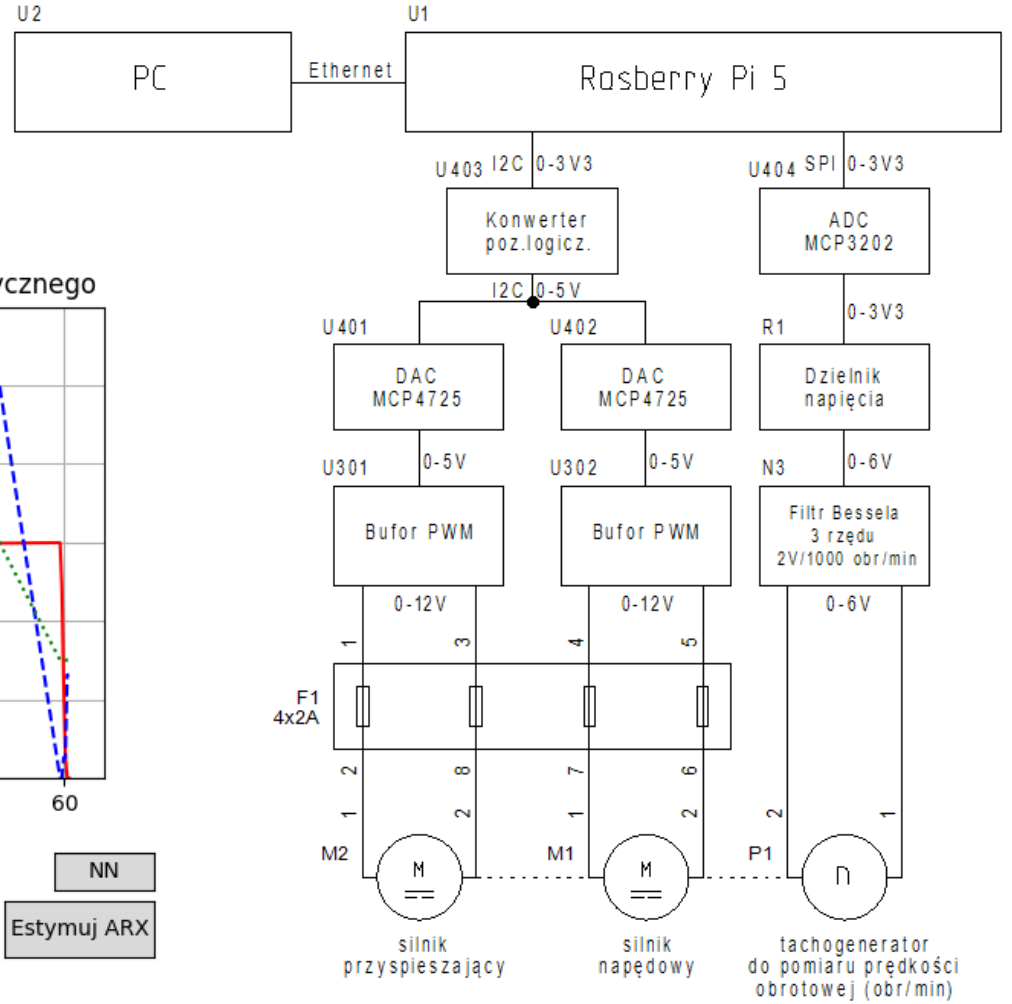
Układ elektryczny i wizualizacja sterowania z poziomu mikrokomputera



Rys. Widok szafy sterowniczej



Rys. Wizualizacja sterowania zestawem napędowym wraz z wyborem metod aktualizowanych nastaw: Zieglera-Nicholsa, Haalmana, Chien-Hrones-Reswick 0%, Chien-Hrones-Reswick 20%, Cohena-Coona, sieci neuronowej



Rys. Schemat blokowy przedstawiający ideę pracy

Opis matematyczny modelu obiektu

Postać transmitancji operatorowej

$$K_i(s) = \frac{k}{\left(1 + \frac{T}{n}s\right)^n}$$

$K_i(s)$ – identyfikowana transmitancja obiektu n-tego rzędu,
k – wzmocnienie,
T – stała czasowa zestawu napędowego,
n – rząd dynamiki.

Postać czasowa odpowiedzi skokowej

$$y(t) = k \cdot \left[1 - e^{-\frac{n}{T}t} \sum_{m=0}^{n} \frac{\left(\frac{n}{T}t\right)^m}{m!}\right]$$

m – krok sumy rzędów modelu,
t – chwila czasu danej próbki.

$$y(t) = k \cdot \left[1 - Q\left(n, \frac{n}{T}t\right)\right], dla\ Q(n,z) = e^{-z} \sum_{m=0}^{n} \frac{z^m}{m!}$$

z – wykładnik funkcji ekspotencjalnej,
Q(n, z) – funkcja gammaincc i jej argumenty.

Zmodyfikowana metoda Strejca

$$J(\theta) = \frac{1}{N} \sum_{i=0}^{N-1} \left(y_i - y_{model}(t_i;\theta)\right)^2, gdzie\ \theta = [k, T, n]$$

N - krok obliczenia sumy błędu średniokwadratowego,
$J(\theta)$ - funkcja kosztu ze zmiennymi,
$\left(y_i - y_{model}(t_i;\theta)\right)^2$ - błąd średniokwadratowy (MSE).

Aproksymacja metodą Skogestada

$$\sum T_s = n\frac{T}{n} = T$$

$$T_1 = T_{10}; T_2 = T_{20} + \frac{T_{30}}{2}; \tau = \tau_0 + \frac{T_{30}}{2} + \sum_{i \geq 4} T_{i0} + \sum_j T_{j0}^{inv} + \frac{h}{2}$$

$$\tau \approx 0 + \frac{T}{2} + (n-3)T + 0 + 0 = \left(n - \frac{5}{2}\right)T$$

T_s - stała czasowa dla aproksymowanego modelu I-rzędu z opóźnieniem,
τ - wartość opóźnienia czasowego aproksymowanego modelu,
T_{n0} - stałe czasowe modelu wyższego rzędu uporządkowane malejąco,
T_{30} - stała czasowa zidentyfikowana,
τ_0 - opóźnienie czasowe modelu wyższego rzędu,
h – czas próbkowania.

Opis matematyczny sieci neuronowej MLP

$$\mathbf{y} = \mathbf{f}(\mathbf{x}, \Delta\boldsymbol{\theta}), \mathbf{f}: \mathbf{x} \in \mathbb{R}^{\mathbf{9}}, \mathbf{y} \in \mathbb{R}^{\mathbf{3}}$$

y - cechy wyjściowe,
x – wektor cech wejściowych,
$\Delta\theta$ – zmiany poszczególnych nastaw regulatora.

$$h_1 = tanh\big(LN(W_1 x + b_1)\big)$$

h_1 - pierwsza warstwa wejściowa,
W_1 - wymiar wagi pierwszej warstwy $W_1 \in \mathbb{R}^{64\times 9}$,
LN - warstwa stabilizująca *LayerNorm*,
b_1 - wektor biasu pierwszej warstwy wejściowej $b_1 \in \mathbb{R}^{64}$.

- mapowanie wszystkich dziewięciu cech wejściowych na 64 cechy wewnętrzne,
- wprowadza ona nieliniowość przeprowadzając detekcję stanu regulacji i zrozumienia cech podanych na wejście.

$$h_2 = tanh\big(LN(W_2 h_1 + b_2)\big)$$

h_2 - druga warstwa ukryta,
W_2 - wymiar wagi drugiej warstw $W_2 \in \mathbb{R}^{64\times 64}$,
b_2 - wektor biasu $b_2 \in \mathbb{R}^{64}$.

- miesza i przekształca w sposób nieliniowy reprezentację warstwy pierwszej,
- umożliwia modelowanie poszczególnych zależności wskaźników całkowych, czasu regulacji i przeregulowania w postaci 64 cech wewnętrznych na kolejne 64 cechy.

$$h_3 = tanh(W_3 h_2 + b_3)$$

h_3 - trzecia warstwa ukryta,
W_3 - wymiar wagi trzeciej warstwy $W_3 \in \mathbb{R}^{32\times 64}$,
b_3 - wektor biasu $b_3 \in \mathbb{R}^{32}$.

- redukuje wymiar, kompresuje wyuczone informacje przechodząc z opisu parametrów obiektu i regulacji do konkretnej decyzji sterującej,
- redukuje 64 cechy wewnętrzne na 32 cechy,
- odpowiada za decyzje korekty nastaw regulatora PID.

$$h_4 = tanh(W_4 h_3 + b_4)$$

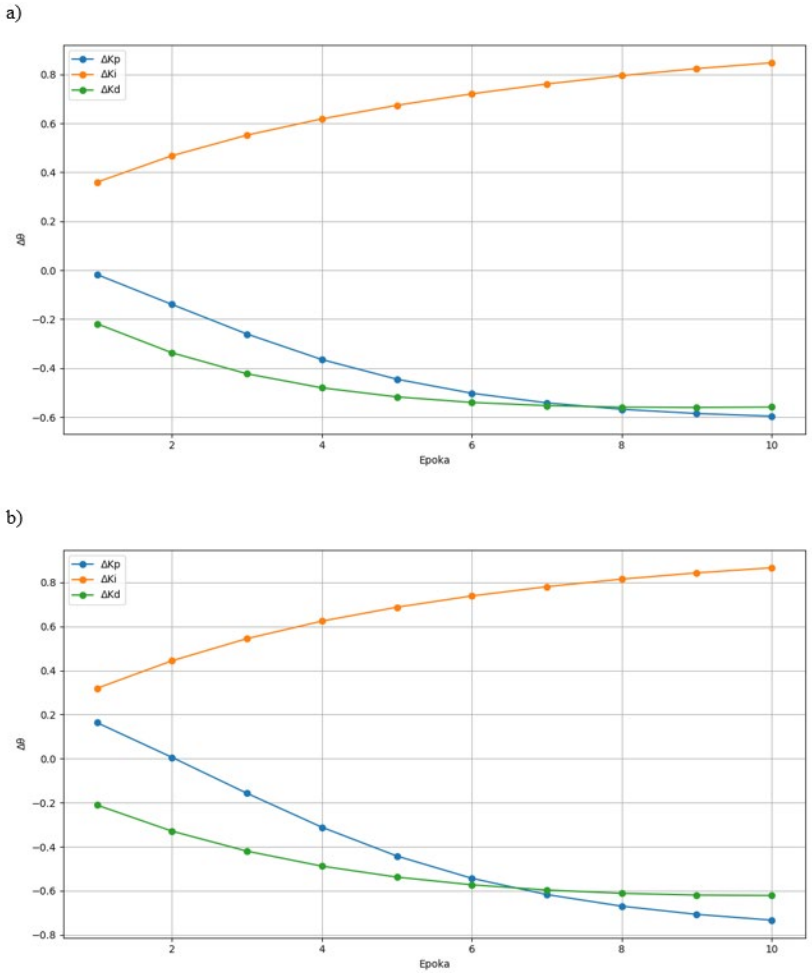
h_4 - czwarta warstwa wyjściowa,
W_4 - wymiar wagi czwartej warstwy $W_4 \in \mathbb{R}^{3\times 32}$,
b_4 - wektor biasu $b_4 \in \mathbb{R}^{3}$.

- zwraca różnicę zmian i kierunku w poszczególnych nastawach części proporcjonalnej, całkującej i różniczkującej,
- redukuje 32 cechy wewnętrzne na 3 cechy określające te zmiany nastaw.

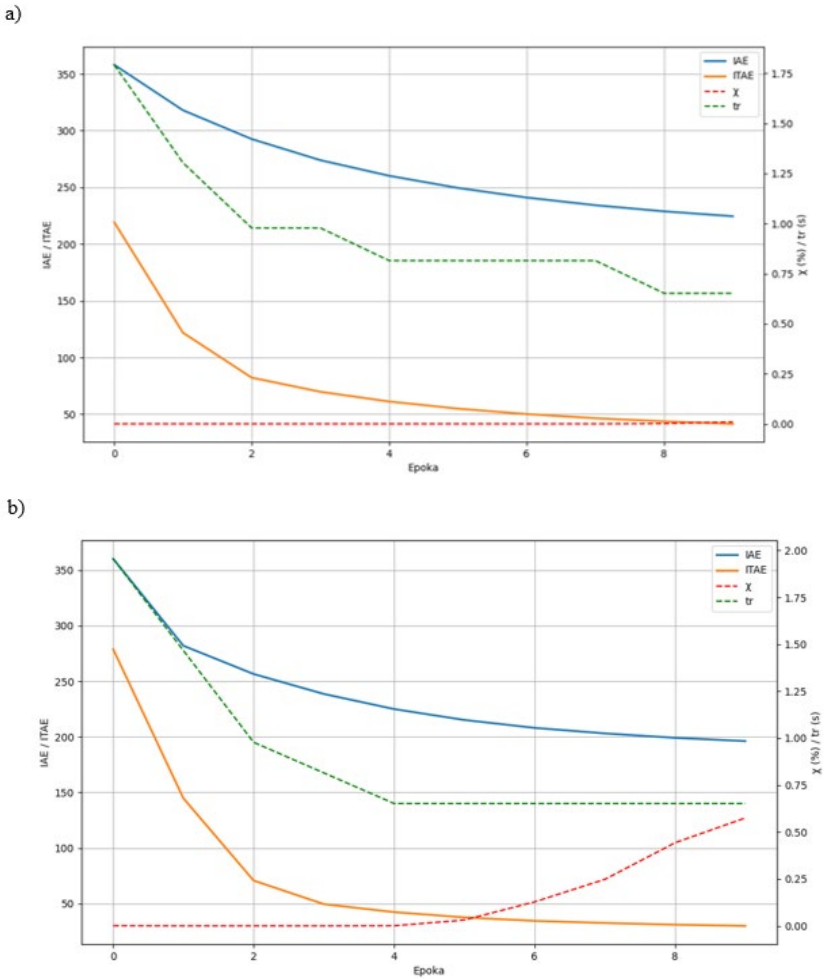
$$\boldsymbol{\theta}_{\mathbf{PID}} = \boldsymbol{\theta}_{\mathbf{0}} + \boldsymbol{\alpha} \cdot \Delta_{\mathbf{NN}}$$

θ_{PID} - wyznaczone nastawy regulatora,
θ_0 - nastawy regulatora bazowe,
α - współczynnik maksymalnej amplitudy korekty (przyjęto 2,3),
Δ_{NN} - wartość korekty.

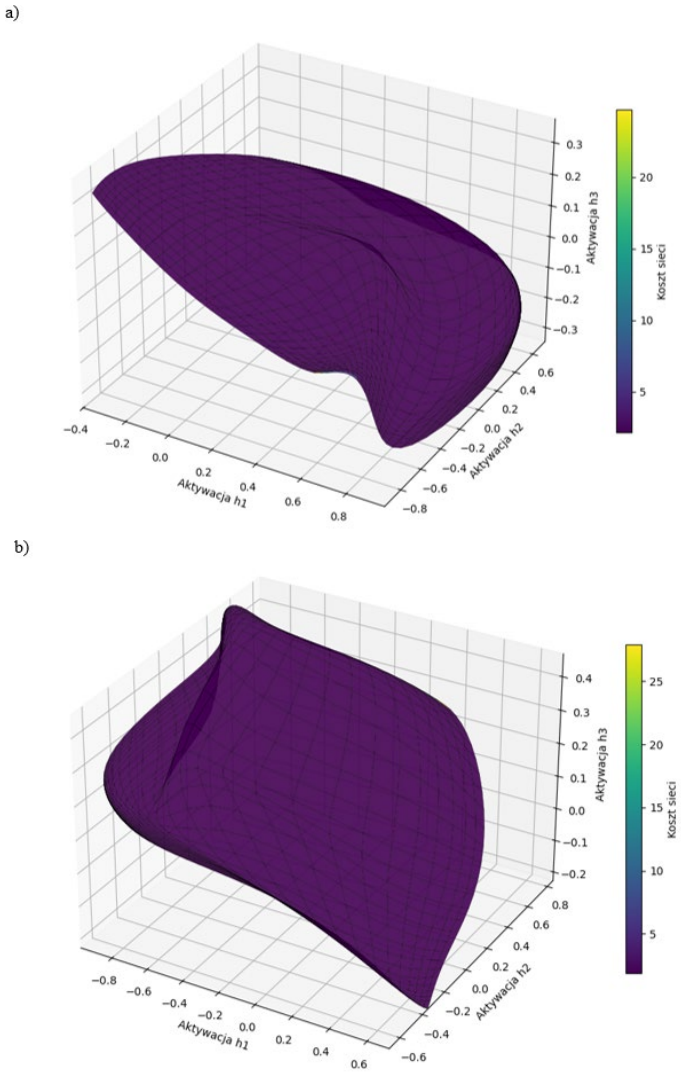
Działanie sieci neuronowej



Rys. Wykres zmian nastaw regulatora PID sieci neuronowej od epok a) bez zakłócenia, b) z zakłóceniem

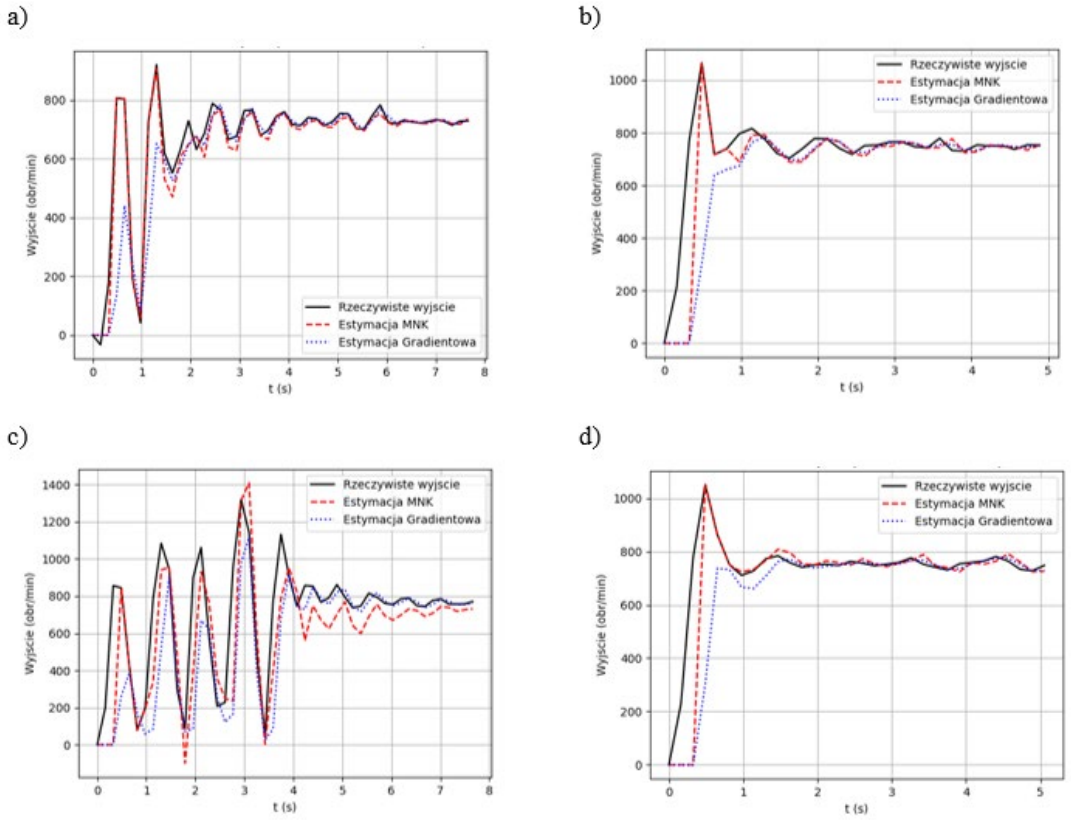


Rys. Zestawienie minimalizacji parametrów kosztu sieci neuronowej: wskaźników całkowych IAE, ITAE, wartości przeregulowania i czasu regulacji a) bez zakłócenia, b) z zakłóceniem

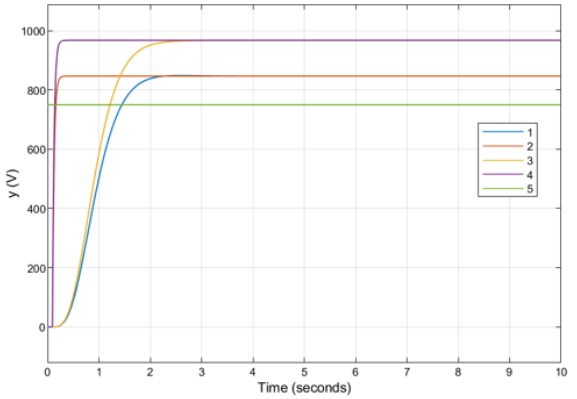


Rys. Wizualizacja przestrzeni działania trzech warstw ukrytych sieci neuronowej a) bez zakłócenia, b) z zakłóceniem

Działanie układu regulacji i symulacja



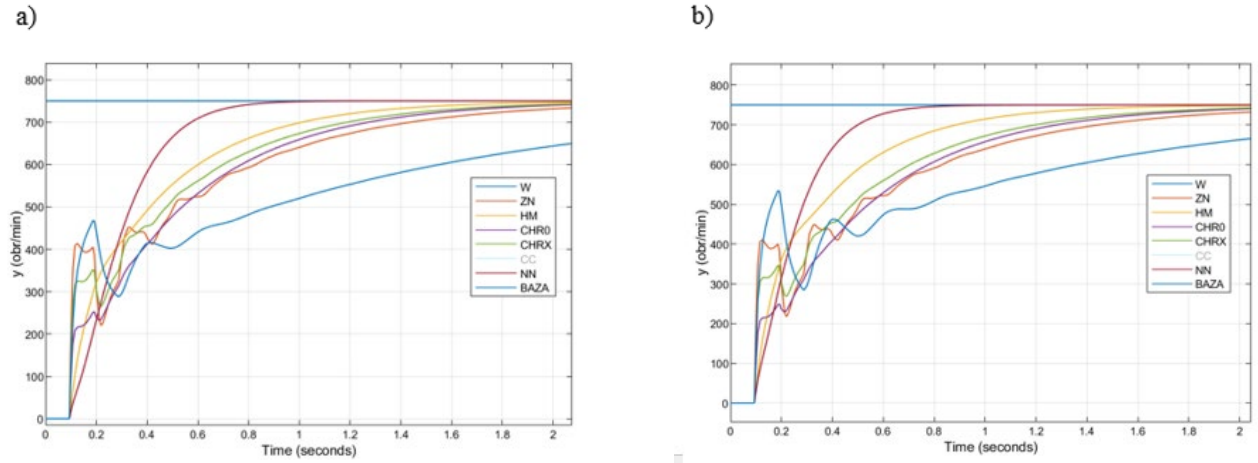
Rys. Wykres estymacji sygnału wyjściowego y metodami MNK i gradientową: a) przed, b) po adaptacji nastaw siecią neuronową bez zakłócenia silnikiem M2 oraz c) przed, d) po adaptacji siecią neuronową z zakłóceniem silnikiem M2



Rys. Wykres symulacji Simulink zestawiający zidentyfikowane i aproksymowane sygnały odpowiedzi skokowej

1 – model n-rzędu bez obciążenia,
2 – model aproksymowany do postaci I-rzędu z opóźnieniem bez obciążenia,
3 – model n-rzędu z obciążeniem,
4 – model aproksymowany do postaci I-rzędu z opóźnieniem z obciążeniem,
5 – wartość zadana.

Metody doboru nastaw regulatorów PID:
W – wartość zadana,
ZN – Zieglera-Nicholsa,
HM – Haalmana,
CHR0 – Chien-Hrones-Reswick 0% przeregulowania,
CHRX - Chien-Hrones-Reswick 20% przeregulowania,
CC – Cohen-Coon,
NN – sieć neuronowa MLP,
BAZA – nastawy podstawowe programu Rasberry Pi



Rys. Wykresy symulacji Simulink odpowiedzi skokowych dla układu regulacji a) bez obciążenia b) z obciążeniem

Wyniki badań

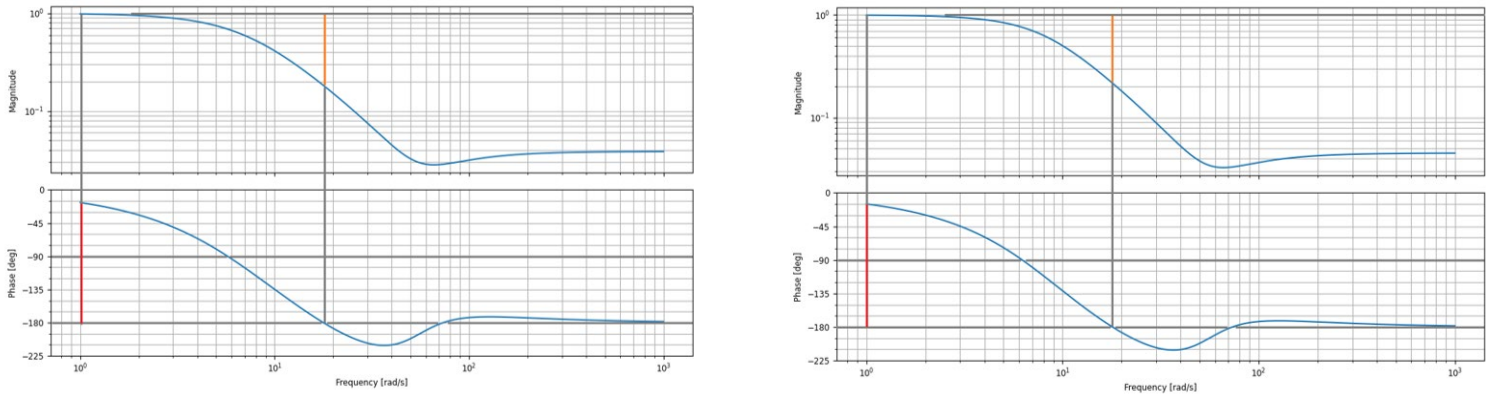
Badania symulacyjne

Metoda doboru nastaw	Bez obciążenia		Z obciążeniem	
	t_r (s)	χ (%)	t_r (s)	χ (%)
Zieglera-Nicholsa	1.60	0.0	1.61	0.0
Haalmana	1.12	0.0	0.98	0.0
Chien-Hrones-Reswick 0%	1.41	0.0	1.42	0.0
Chien-Hrones-Reswick 20%	1.32	0.0	1.33	0.0
Bazowe (Rasberry Pi)	3.36	0.0	3.02	0.0
Sieć neuronowa MLP	0.61	0.0	0.54	0.0

Obiekt rzeczywisty

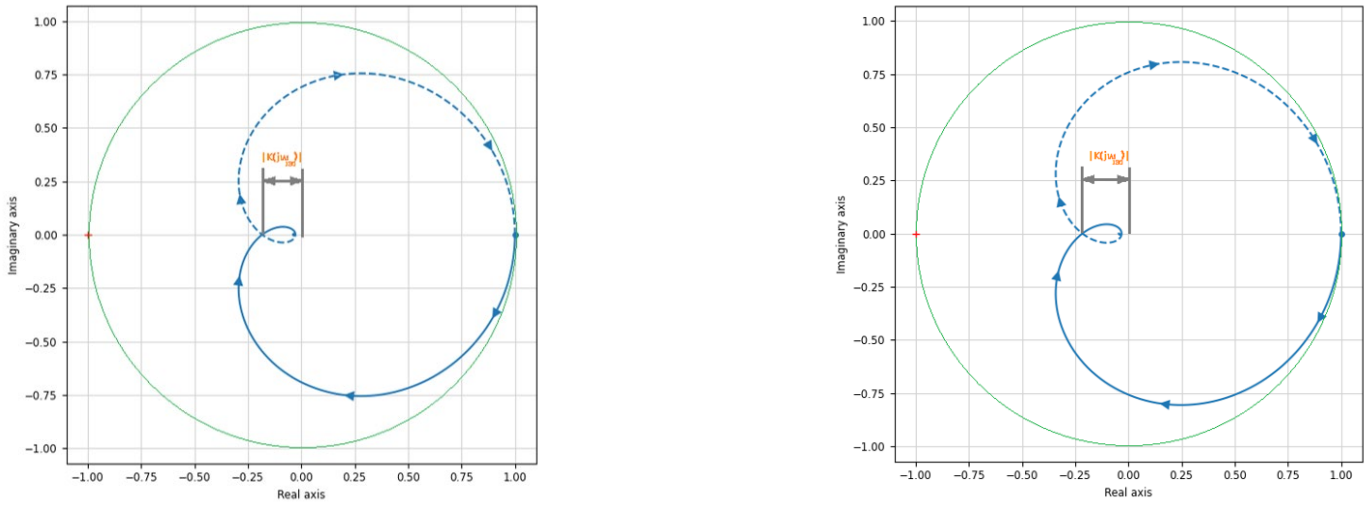
Metoda doboru nastaw	Bez obciążenia		Z obciążeniem	
	t_r (s)	χ (%)	t_r (s)	χ (%)
Zieglera-Nicholsa	1.25	1.9	1.44	1.5
Haalmana	1.16	2.0	0.86	3.4
Chien-Hrones-Reswick 0%	1.29	2.0	1.04	1.7
Chien-Hrones-Reswick 20%	1.11	1.3	0.98	0.9
Bazowe (Rasberry Pi)	3.52	1.5	3.22	0.0
Sieć neuronowa MLP	1.07	6.9	0.97	26.6

a) b)



Rys. Charakterystyka częstotliwościowa logarytmiczna Bodego dla obiektu z nastawami sieci neuronowej a) bez zakłócenia, b) z zakłóceniem (opracowanie własne); **Barwa pomarańczowa** – zapas amplitudy, **Barwa czerwona** – zapas fazy.

a) b)



Rys. Charakterystyka Nyquista do badania stabilności z nastawami sieci neuronowej a) bez zakłócenia, b) z zakłóceniem (opracowanie własne); **Barwa zielona** – okrąg jednostkowy, **Barwa ciemnoszara** – zapas amplitudy.

System regulacji adaptacyjnej z wykorzystaniem sieci neuronowej MLP pozwala na osiągnięcie:

- ✓ bardzo zadowalających wyników:
  - najlepszych czasów regulacji i jakości sterowania,
  - charakterystyk Bodego i Nyquista wykazujących na duży zapas amplitudy, fazy i stabilności układu regulacji,
  - minimalnego przeregulowania kosztem wyeliminowania oscylacji podczas czasu narastania,
- ✓ spektrum działania sieci neuronowej blisko **minimalnego kosztu**,
- ✓ wystarczającej bazy danych dla **cech wejściowych sieci neuronowej** na podstawie podejścia identyfikacji zmodyfikowaną metodą Strejca, aproksymacji metodą Skogestada oraz estymacji metodą MNK,
- ✓ minimalnych różnic w czasach regulacji w zestawieniu wyników z symulacją metodą *Model-in-the-loop* i rzeczywistego układu,
- ✓ **najlepsza metoda**: Haalmana oraz sieci neuronowej MLP.

Zastosowanie **sztucznej inteligencji** w praktyce inżynierskiej pozwala na:

- ✓ otrzymywanie lepszych rezultatów niż standardowe metody doboru nastaw regulatorów,
- ✓ pozwala zoptymalizować procesy regulacji w przemyśle,
- ✓ zmniejszyć wpływ przed procesami starzeniowymi, zakłóceniami, czy zmianami obciążenia,
- ✓ przewidywać awarie i zachowanie układów regulacji, których bezpośrednia interpretacja matematyczna jest utrudniona.