PROJEKT SPECJALNOŚCIOWY
Algorytm sterowania platformą klasy (2,0)

Autor: Emilia Szymańska 248975
czw 16:10

15.11.2021r.

1 Wstęp

Platforma klasy (2,0) - inaczej zwana monocyklem - jest układem nieholonomicznym z dwoma kołami napędowymi, bez żadnych kół sterowalnych. Układ taki można opisać przy pomocy kinematyki (ograniczeń nieholonomicznych) oraz dynamiki (we współrzędnych pomocniczych).

W przypadku bezdryfowego układu sterowania mamy do czynienia z postacią:

$$\dot{q} = G(q) \cdot \eta, \tag{1}$$

gdzie:

- $q \in \mathbb{R}^n$ - wektor uogólnionych współrzędnych,
- $\dot{q}$ - wektor uogólnionych prędkości,
- $\eta \in \mathbb{R}^{n-l}$ - wektor prędkości pomocniczych,
- $G(q)$ - macierz o wymiarze nx$(n-l)$ opisująca właściwości platformy,
- n - wymiar stanu współrzędnych uogólnionych,
- l - liczba ograniczeń nieholonomicznych.

Po rozwinięciu, równania kinematyki przyjmują następującą postać:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\varphi}_1 \\ \dot{\varphi}_2 \end{bmatrix} = \begin{bmatrix} \cos\theta & \cos\theta \\ \sin\theta & \sin\theta \\ \frac{1}{L} & -\frac{1}{L} \\ \frac{2}{R} & 0 \\ 0 & \frac{2}{R} \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix}, \tag{2}$$

gdzie:

- $\dot{x}$ - prędkość liniowa w osi X,
- $\dot{y}$ - prędkość liniowa w osi Y,
- $\dot{\theta}$ - prędkość kątowa ustawienia platformy,
- $\dot{\varphi_1}$, $\dot{\varphi_2}$ - prędkości kątowe kół,
- R - promień koła napędowego,
- L - długość platformy,
- η_1, η_2 - prędkości pomocnicze kół.

Z tych zależności można wyprowadzić, że prędkość liniowa platformy określona jest wzorem $\eta_1 + \eta_2 = v$, natomiast jej prędkość kątową można wyrazić jako $\eta_1 - \eta_2 = \omega$.

By otrzymać układ dosterowany i pozbyć się mnożników Lagrange'a λ, przekształcenia zostały dokonane na równaniu dynamiki:

$$M \cdot \ddot{q} + C \cdot \dot{q} + D = B \cdot u + A \cdot \lambda \tag{3}$$

$$M \cdot \left(\dot{G}\eta + G\dot{\eta}\right) + C \cdot G\eta + D = B \cdot u + A \cdot \lambda \tag{4}$$

$$G^T MG \cdot \dot{\eta} + G^T \left(M\dot{G} + CG\right) \eta + G^T D = G^T B \cdot u + G^T A^T \cdot \lambda \tag{5}$$

$$M^* \cdot \dot{\eta} + C^* \cdot \eta + D^* = B^* \cdot u \tag{6}$$

W tym przypadku macierz inercji M, macierz sił Coriolisa i odśrodkowych bezwładności C, wektor grawitacji, macierz Pfaffa A, wektor sił grawitacji D oraz prostokątną macierz wejściową B zastępujemy nowymi postaciami odpowiadających macierzy i wektorów M^*, C^*, D^* i B^*.

2 Sterownik kinematyczny

Celem sterowania, który rozpatrujemy, jest śledzenie trajektorii. Kinematyka robota opisana jest następującymi równaniami:

$$\dot{x} = v \cdot \sin\theta \tag{7}$$

$$\dot{y} = v \cdot \cos\theta \tag{8}$$

$$\dot{\theta} = \omega \tag{9}$$

zatem najprostszą dopuszczalną (spełniającą wymagania nieholonomiczne) trajektorię zadaną można wyrazić w następujący sposób:

$$\dot{x_d} = v_d \cdot \sin\theta_d \tag{10}$$

$$\dot{y_d} = v_d \cdot \cos\theta_d \tag{11}$$

$$\dot{\theta_d} = \omega_d \tag{12}$$

gdzie indeks d wskazuje na wartość zadaną, a $v_d = r \cdot \omega_d$ (r - promień okręgu, po którym ma poruszać się robot).

Wyróżniamy błędy podstawowe śledzenia trajektorii:

$$x_e = x - x_d \tag{13}$$

$$y_e = y - y_d \tag{14}$$

$$\theta_e = \theta - \theta_d \tag{15}$$

oraz błędy referencyjne:

$$\begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = Rot(Z, -\theta) \begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix}. \tag{16}$$

Po przekształceniach otrzymujemy:

$$\begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} \cos\theta\,(x - x_d) + \sin\theta\,(y - y_d) \\ -\sin\theta\,(x - x_d) + \cos\theta\,(y - y_d) \\ \theta - \theta_d \end{bmatrix}. \tag{17}$$

Pochodne błędów referencyjnych mają postać równań kaskadowych:

$$\dot{e_x} = \omega e_y + v - v_d \cos e_\theta \tag{18}$$

$$\dot{e_y} = -\omega e_x + v_d \sin e_\theta \tag{19}$$

$$\dot{e_\theta} = \omega - \omega_d \tag{20}$$

Do rozwiązania układu kaskadowego, potrzebne było zdefiniowanie funkcji Lapunowa:

$$V\left(e_x, e_y, e_\theta\right) = \frac{1}{2}e_x^2 + \frac{1}{2}e_y^2 + \frac{1}{2}e_\theta^2. \tag{21}$$

Funkcja ta jest zawsze większa lub równa 0. Pochodna tej funkcji w postaci:

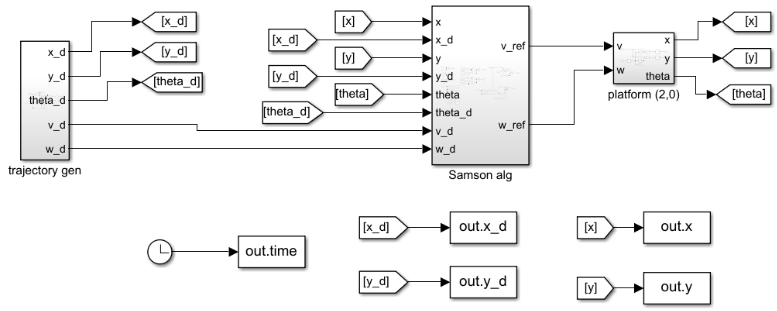
$$\dot{V}\left(e_x, e_y, e_\theta\right) = -k_1 e_x^2 - k_2 e_\theta^2 \tag{22}$$

dla $k_1, k_2 > 0$ jest zawsze mniejsza lub równa zero. Na tej podstawie wyznaczone zostały prędkości referencyjne wykorzystywane w algorytmie Samsona:

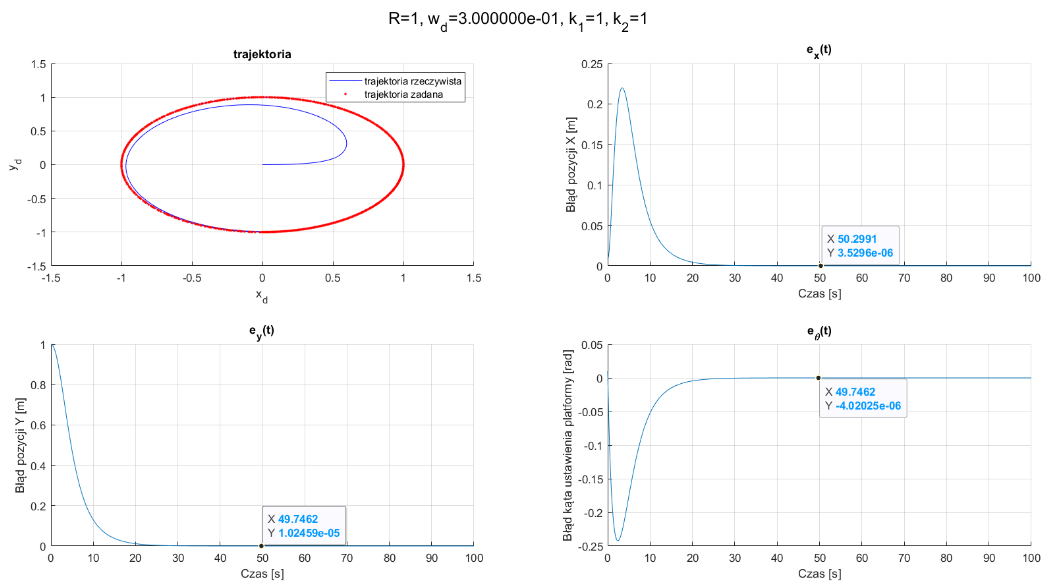
$$v_{ref} = v_d \cos e_\theta - k_1 e_x \tag{23}$$

$$\omega_{ref} = \omega_d - \frac{e_y v_d \sin e_\theta}{e_\theta} - k_2 e_\theta \tag{24}$$

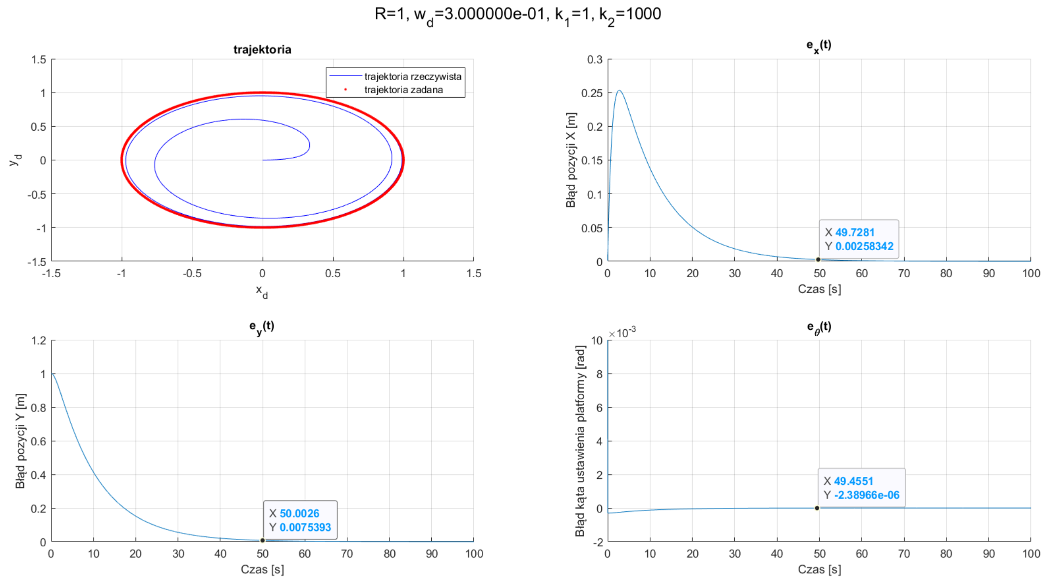
Zaimplementowany system z algorytmem Samsona został przedstawiony na Rys. 1. Następnie seria badań wpływu parametrów na działanie została przeprowadzona. W porównaniu do sytuacji z $k_1 = 1$, $k_2 = 1$ (rys. 2), przypadki zwiększające jeden z tych parametrów do wartości 1000 sprawia, że więcej czasu zajmuje platformie osiągnięcie zamierzonej trajektorii (Rys. 3, 4). Ostatni zbadany scenariusz zakładał oba parametry na poziomie wartości 1000 - wówczas platforma nie zdąża w 100 sekund osiągnąć swojej trajektorii (Rys. 5).



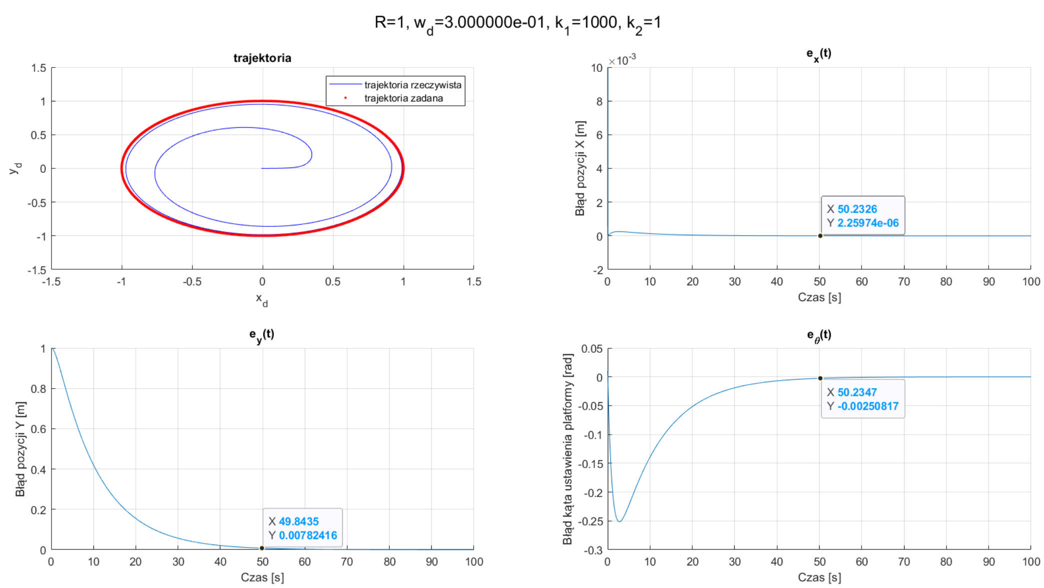
Rysunek 1: Schemat systemu w Simulinku, który implementuje działanie algorytmu kinematycznego.



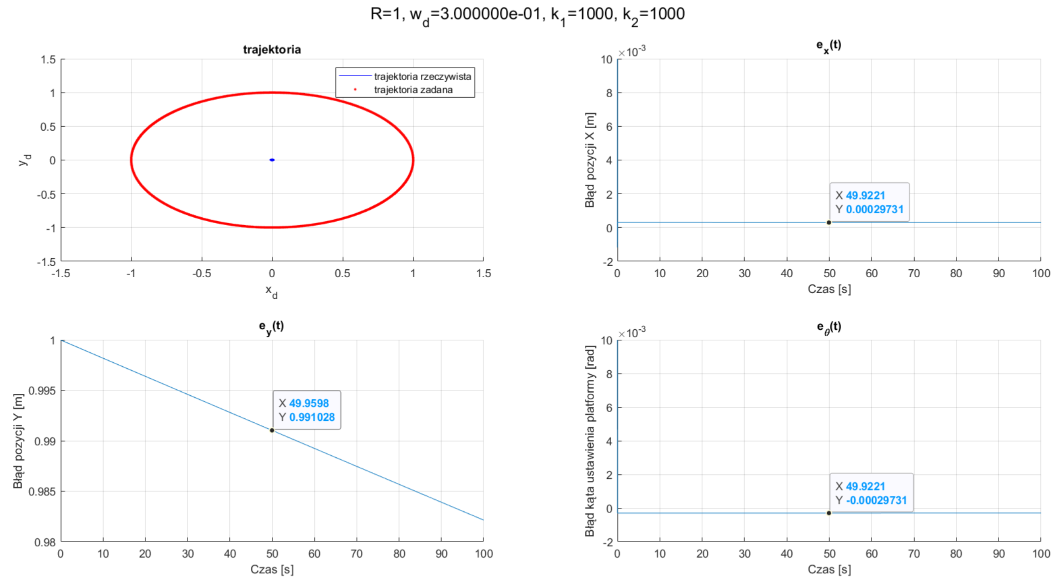
Rysunek 2: Błędy i trajektoria platformy dla algorytmu Samsona z parametrami $k_1 = 1$, $k_2 = 1$.



Rysunek 3: Błędy i trajektoria platformy dla algorytmu Samsona z parametrami $k_1 =$, $k_2 = 1000$.



Rysunek 4: Błędy i trajektoria platformy dla algorytmu Samsona z parametrami $k_1 = 1000$, $k_2 = 1$.



Rysunek 5: Błędy i trajektoria platformy dla algorytmu Samsona z parametrami $k_1 = 1000$, $k_2 = 1000$.

3 Sterownik dynamiczny

Dynamika platformy (2,0) opisać można następującym układem równań:

$$M^* \dot{\eta} = u. \tag{25}$$

Macierz M^* można przedstawić w formie:

$$\begin{bmatrix} M_{11} & 0 \\ 0 & M_{22} \end{bmatrix}, \tag{26}$$

gdzie:

- $M_{11} \approx 107,9$ kg – całkowita masa platformy,
- $M_{22} \approx 83,6$ kg – iloraz momentu bezwładności platformy przez kwadrat długości platformy.

Dokonywane są przekształcenia, by móc wyrazić obiekt z kinematyką i dynamiką zaimplementowaną w Simulinku:

$$\dot{\eta} = M^{*-1} u, \tag{27}$$

$$\begin{bmatrix} \dot{v} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \frac{1}{M_{11}} & 0 \\ 0 & \frac{1}{M_{22}} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \tag{28}$$

$$\dot{v} = \frac{1}{M_{11}} u_1 \tag{29}$$

$$\dot{\omega} = \frac{1}{M_{22}} u_2. \tag{30}$$

Po zaimplementowaniu obiektu należy przejść do kwestii samego sterownika. Sterowanie odbywa się za pomocą algorytmu dokładnej linearyzacji, którego zasadę działania prezentuje wzór:

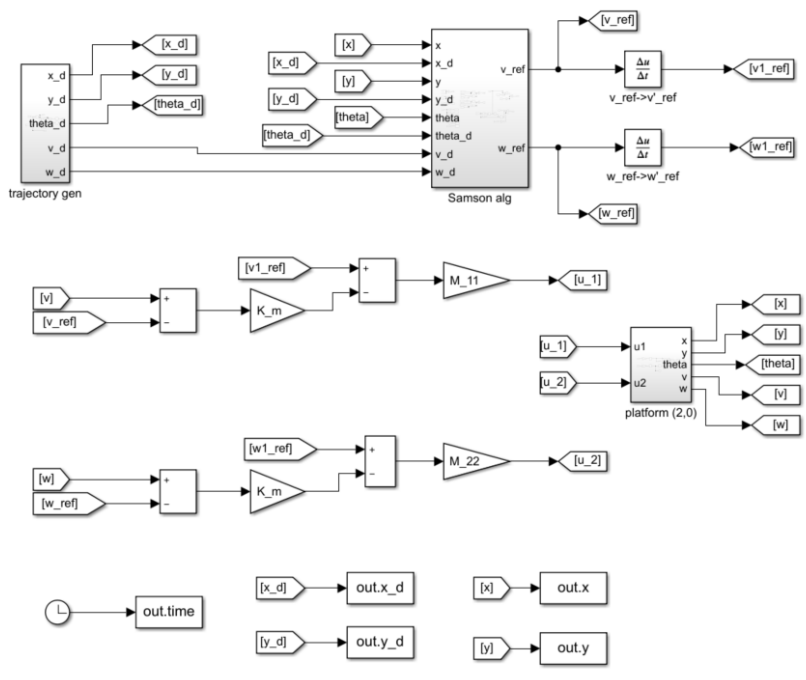
$$u = M^* \left(\dot{\eta}_{ref} - K_m \left(\eta - \eta_{ref}\right)\right). \tag{31}$$

Po rozwinięciu:

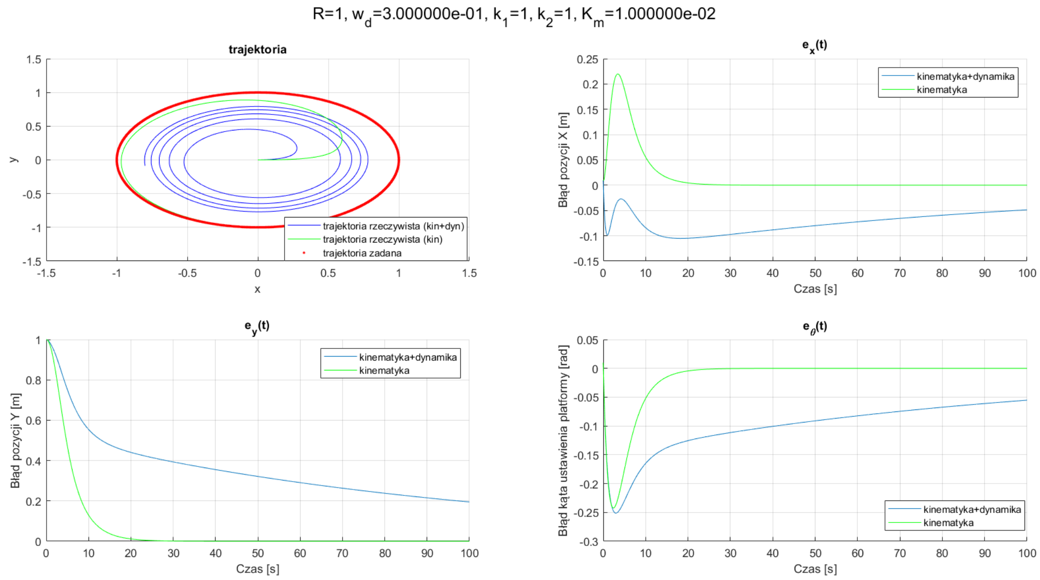
$$u_1 = M_{11} \left(\dot{v}_{ref} - K_m \left(v - v_{ref}\right)\right) \tag{32}$$

$$u_2 = M_{22} \left(\dot{\omega}_{ref} - K_m \left(\omega - \omega_{ref}\right)\right). \tag{33}$$

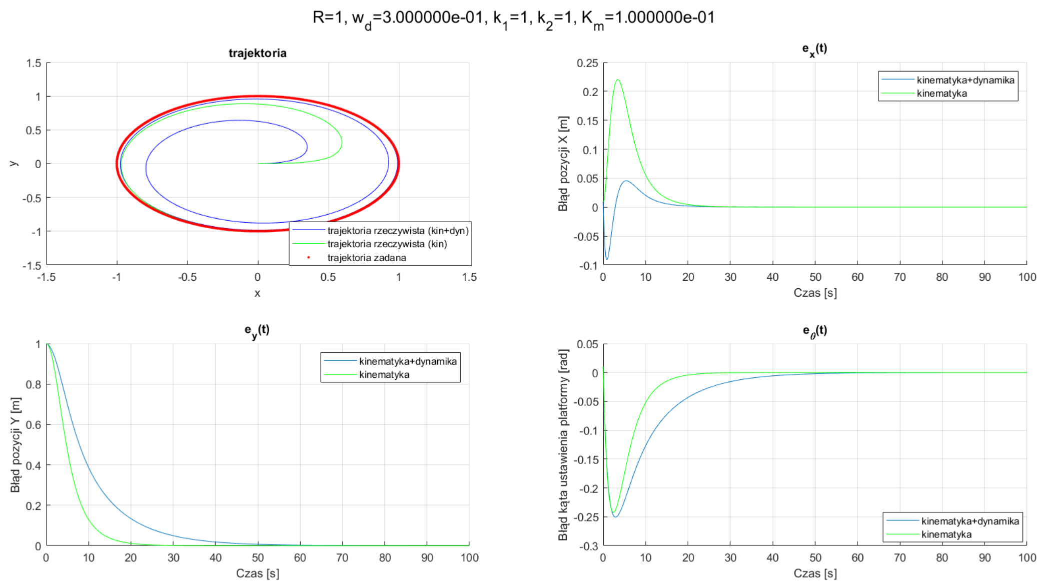
Wartości η_{ref} pobierane są ze sterownika kinematycznego, a współczynnik K_m badany w zadaniu musi być dodatni. Na podstawie tych wyprowadzeń został zbudowany system jak na Rys. 6. Zbadane zostały trzy przypadki zmiany wartości parametru K_m: $K_m = 0.01$ (Rys. 7), $K_m = 0.1$ (Rys. 8), $K_m = 0.01$ (Rys. 7). Można zauważyć, że im większy parametr K_m, tym lepsze działanie sterownika - trajektoria w przypadku uwzględnionej w sterowaniu kinematyki i dynamiki jest coraz bliższa przypadkowi implementacji samej kinematyki obiektu. Dodatkowa różnica jest zauważalna przy różnych współczynnikach i warunkach początkowych (Rys.10), szczególnie przy zmianie parametrów k_1 i k_2 - robot albo się zbliża od zewnątrz do zadanej trajektorii, albo przecina ją i dąży do niej od wewnątrz.



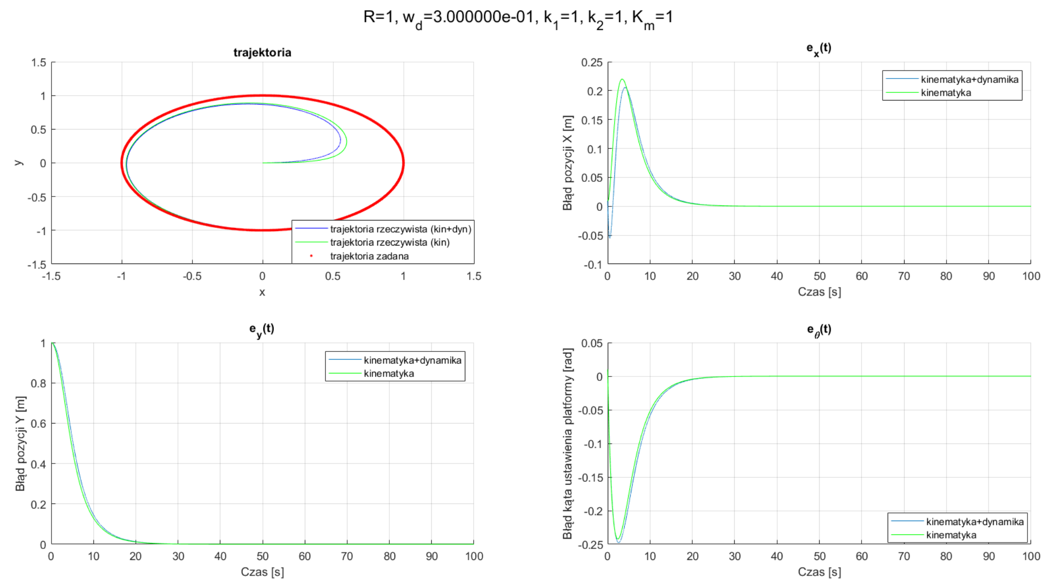
Rysunek 6: Schemat systemu w Simulinku, który implementuje działanie algorytmu kinematycznego i dynamicznego.



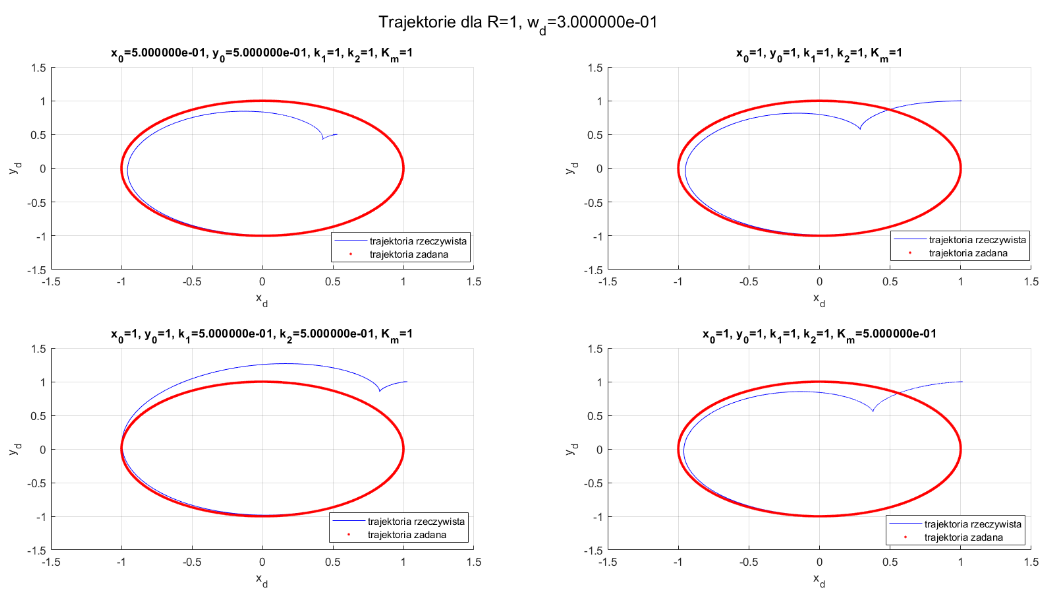
Rysunek 7: Błędy i trajektoria platformy dla algorytmu dokładnej linearyzacji z parametrami $k_1 = 1$, $k_2 = 1$, $K_m = 0.01$.



Rysunek 8: Błędy i trajektoria platformy dla algorytmu dokładnej linearyzacji z parametrami $k_1 = 1$, $k_2 = 1$, $K_m = 0.1$.



Rysunek 9: Błędy i trajektoria platformy dla algorytmu dokładnej linearyzacji z parametrami $k_1 = 1$, $k_2 = 1$, $K_m = 1$.



Rysunek 10: Trajektoria platformy dla algorytmu dokładnej linearyzacji z różnymi współczynnikami i wartościami początkowymi.

4 Wnioski

- Dodanie sterownika dynamicznego (z odpowiednim współczynnikiem) do obiektu o właściwościach kinematycznych i dynamicznych pozwala wymusić na obiekcie pożądane prędkości, a tym samym poprawia szybkość osiągania zadanej trajektorii.
- Im większy współczynnik K_m w sterowniku dynamicznym, tym bardziej trajektoria rzeczywista jest zbliżona do trajektorii obiektu sterowanego sterownikiem kinematycznym.
- Zwiększanie parametrów k_1 i k_2 w przypadku sterownika kinematycznego wydłuża czas osiągania zadanej trajektorii.
- W zależności od wartości początkowych i współczynników sterowników, można osiągnąć zbliżanie się do zadanej trajektorii od zewnątrz lub – po przecięciu trajektorii – od wewnątrz.