

POLITECHNIKA WROCŁAWSKA  
WYDZIAŁ ELEKTRONIKI

---

KIERUNEK: Automatyka i Robotyka (AIR)  
SPECJALNOŚĆ: Robotyka (ARR)

**PRACA DYPLOMOWA  
MAGISTERSKA**

Zastosowanie algorytmów sterowania z poślizgami  
do robota kroczącego z nogami typu wheg

Skid-steering algorithm for hexapod wheg mobile  
robot control

AUTOR:  
Bartosz Witkowski

PROWADZĄCY PRACĘ:  
dr inż. Robert Muszyński

OCENA PRACY:



# Spis treści

<b>1</b>	<b>Wstęp</b>	<b>3</b>
<b>2</b>	<b>Model robota</b>	<b>7</b>
2.1	Zasada d'Alemberta . . . . .	7
2.2	Model we współrzędnych uogólnionych . . . . .	8
2.3	Model we współrzędnych pomocniczych . . . . .	9
2.4	Model kinematyki . . . . .	10
2.4.1	Model z dopuszczonymi poślizgami poprzecznymi . . . . .	11
2.5	Model dynamiki robota trójkołowego . . . . .	12
2.6	Parametry badanej konstrukcji . . . . .	14
<b>3</b>	<b>Algorytmy sterowania robota</b>	<b>15</b>
3.1	Prosty algorytm ruchu po krzywiźnie . . . . .	15
3.2	Pomocniczy algorytm sterowania do punktu . . . . .	16
3.3	Pełny algorytm sterowania do punktu . . . . .	17
<b>4</b>	<b>Badania symulacyjne</b>	<b>19</b>
4.1	Model robota trójkołowego . . . . .	19
4.1.1	Weryfikacja modelu poślizgów bocznych . . . . .	19
4.2	Sterowanie do punktu . . . . .	19
<b>5</b>	<b>Podsumowanie</b>	<b>23</b>
	<b>Bibliografia</b>	<b>23</b>



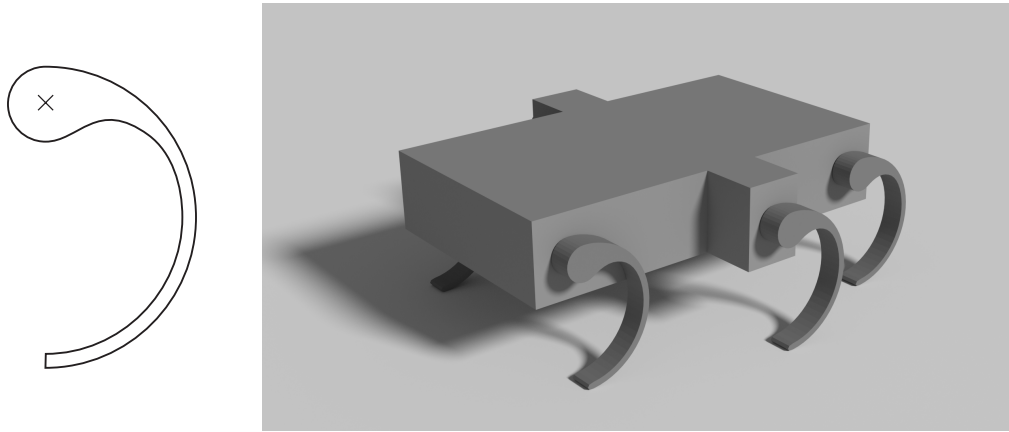
# Rozdział 1

## Wstęp

W drugiej połowie lat 90-tych XX wieku powstały pierwsze roboty kroczące wykorzystujące napęd typu whег, m. in. RHex [1]. Roboty z napędem tego typu budową znacząco odbiegają od większości konstrukcji kroczących [2]. Podstawą napędu robota są nogi typu whег. Są one połączeniem koncepcji napędzanego koła oraz klasycznej nogi. Pozwalają one na zastosowanie jednego napędu na nogę, podczas gdy najpopularniejsze konstrukcje kroczące wykorzystują aż trzy napędy. Zastosowanie tego typu nóg pozwala zatem znacząco zmniejszyć liczbę napędów, a co za tym idzie koszty oraz stopień skomplikowania konstrukcji mechanicznej. Skutkuje to jednak koniecznością wykorzystania znacznie bardziej złożonego algorytmu sterowania.

Napęd typu whег zapewnia zdolność poruszania się w trudnym terenie zbliżoną do uzyskiwanej przez klasyczne roboty kroczące przy zachowaniu prostej budowy mechanicznej. Pozwala on na pokonywanie przeszkód niemożliwych do pokonania przez roboty kołowe, np. zazwyczaj (zależnie od kształtu nóg) możliwe jest wchodzenie po schodach oraz przekroczenie przeszkody o wysokości sięgającej długości nogi (znacznie wyższej niż dla robotów kołowych). Spotykane są implementacje napędu typu whег różniące się między sobą kształtem nogi. Pierwsze nogi tego typu były proste, z punktem obrotu umieszczonym na jednym z końców. Często zakończone były elementem mającym zapewnić lepszy kontakt z podłożem. Obecnie praktycznie nie stosuje się tego typu nóg, zostało ono wyparte przez bardziej złożone rozwiązania. Innym podejściem jest umieszczenie kilku prostych, promieniście rozłożonych ramion na pojedynczym napędzie. Sterowanie takim robotem jednak niemal nie różni się od sterowania klasycznym robotem kołowym.

Inspiracją dla niniejszej pracy jest robot zbudowany w ramach projektu zespołowego na Politechnice Wrocławskiej w 2014 roku [3]. Jego zewnętrzne wymiary (pomijając nogi) to 350 mm długości, 300 mm szerokości i 70 mm wysokości. Głównym elementem konstrukcyjnym jest rama aluminiowa. Całość waży około 3kg. Nogi robota wykonano technologią druku FDM z polilaktydu. Mają one kształt półokręgu o średnicy 12 cm i zostały zamocowane do napędów w sposób zaprezentowany na rysunku 1.1. Rozwiązanie to pozwala m. in. na płynniejsze podnoszenie się robota z ziemi oraz łagodniejsze przejście nogi z fazy przenoszenia do fazy podparcia względem rozwiązań z prostymi nogami. Zwiększona jest również powierzchnia kontaktu nogi z podłożem, co może mieć kluczowe znaczenie przy poruszaniu się po trudniejszym terenie. Napęd robota stanowi 6 silników DC z przekładniami oraz enkoderami. Maksymalna prędkość obrotowa silników wraz z przekładniami wynosi 200 obr/min., a maksymalny moment obrotowy to 1,2Nm. Enkodery zamocowane są bezpośrednio na wale silnika, co w połączeniu z przekładnią pozwala na pomiar położenia nogi z dokładnością do  $0,1125^\circ$ . W robocie zaimplementowane zostały zarówno regulatory pozycji jak i prędkości obrotowej dla każdej z nóg.



Rysunek 1.1 Schemat kształtu nogi oraz wizualizacja badanego robota

Jak wspomniano z prostą budową mechaniczną robotów o typu whég wiąże się niestety konieczność projektowania nietypowych algorytmów sterowania. Większość prac dotyczących robotów tego typu traktuje o ich konstrukcji mechanicznej [4], nieliczne o sposobie sterowania [5] [6], żadna o sterowaniu w oparciu o model kinematyki czy dynamiki. Uprozczone postępowanie często nie daje dobrych rezultatów przez pominięcie dynamiki, w tym poślizgów, które grają znaczną rolę w reakcji robota na sterowania. Stąd przedmiotem niniejszej pracy jest propozycja sposobu postępowaniaprowadzącego do uzyskania algorytmu sterowania robota do punktu, uwzględniającego model jego kinematyki i dynamiki.

Algorytmy chodu często rozróżniane są ze względu na liczbę nóg mających jednocześnie kontakt z podłożem. W przypadku robota sześcionożnego warto wspomnieć o chodach trzy-, cztero- i pięciopodporowym, odpowiadających trzem, czterem i pięciu nogom w fazie podparcia. Ze względu na ograniczenia wynikające z zastosowania napędu typu whég w pracy rozważać będziemy jedynie chód trójpodporowy. Możliwe byłoby również wykorzystanie chodu czteropodporowego, jednak przy napędzie typu whég ma on zastosowanie jedynie przy przekraczaniu przeszkód.

Podczas projektowania algorytmów chodu istotnym czynnikiem jest stabilność chodu. Przyjmuje się, że chód jest stabilny wówczas, gdy rzut środka ciężkości robota na płaszczyznę znajduje się wewnątrz wielokąta podparcia definiowanego przez punkty, w których nogi mają kontakt z podłożem. Minimalna odległość tego punktu od krawędzi wielokąta jest nazywana zapasem stabilności. Warunek ten daje gwarancję, że robot w bezruchu nie przewróci się. Ważny jest więc prawidłowy dobór nóg mających kontakt z podłożem a także wzajemna synchronizacja ich ruchu.

Dla ustalenia nomenklatury warto zdefiniować pojęcia fazy przenoszenia (protrakcji) oraz fazy podparcia (retrakcji) [7]. Faza przenoszenia jest to okres, w którym dana noga nie ma kontaktu z podłożem i jest przemieszczana w celu dotarcia do punktu, w którym zostanie ona postawiona. Wówczas noga przechodzi w fazę podparcia, która trwa do momentu oderwania jej od podłoża i rozpoczęcia przenoszenia. Pełen cykl przejścia przez fazę podparcia i przenoszenia nazywany jest okresem chodu.

Podstawą do opracowania algorytmu sterowania robota z nogami typu whég w oparciu o jego model dynamiki stanowi obserwacja, iż w chodzie trójpodporowym taki robot może być chwilowo traktowany jak robot trójkołowy z nieskrętnymi kołami zmieniającymi swoje położenie w momencie zmiany nóg, które są w fazie podparcia. Jasne jest, że aby umożliwić skręcanie tego typu robota należy dopuścić występowanie poślizgów w trakcie

---

tego manewru. Obserwacje te pozwalają na adaptację istniejących algorytmów sterowania robotami kołowymi z uwzględnieniem poślizgów [8] i stanowi główny pomysł pracy.

Celem pracy jest wyprowadzenie modelu kinematyki i dynamiki robota typu whæg z uwzględnieniem częściowych poślizgów oraz wykorzystanie go w zadaniu sterowania do punktu. Do rozwiązania zadania sterowania do punktu zostanie wykorzystana metoda endogenicznej przestrzeni konfiguracyjnej. Przeprowadzone zostaną symulacyjne mające na celu zbadanie własności algorytmu sterowania.

Układ pracy jest następujący. Rozdział 2 przedstawia teoretyczne podstawy, które wykorzystano do wykonania modelu robota a także modele kinematyki i dynamiki. Rozdział 3 poświęcono algorytmowi sterowania robota zarówno na prostym poziomie sterowania prędkościami jak i sterowania do punktu z uwzględnieniem modelu dynamiki. Uwzględniono także sterowanie modelem przełączanym. W rozdziale 4 umieszczono wyniki badań symulacyjnych wraz z ich omówieniem i interpretacją. Rozdział 5 jest podsumowaniem wykonanych prac.





# Rozdział 2

## Model robota

Zgodnie ze sposobem postępowania nakreślonym we wstępie na potrzeby pracy wyprowadzony zostanie model trójkołowego robota mobilnego z nieskrętnymi kołami. Istnieją dobrze znane metody wyprowadzania modelu kinematyki i dynamiki kołowych robotów mobilnych zarówno przy założeniu braku poślizgów jak i przy dopuszczeniu poślizgów częściowych [9]. Tutaj interesować nas będzie drugi przypadek i to on zostanie opracowany.

Przez kinematykę nieholonomicznego układu rozumie się zbiór reguł opisujących wzajemne zależności zmiennych stanu. Zatem kinematyka robota to równania ograniczeń. Model kinematyki pozwala na bezpośrednią kontrolę prędkości celem zrealizowania zadania sterowania. Mając jednak do czynienia z rzeczywistym robotem nie jest to możliwe, gdyż sterujemy silnikami, które generują siły mające dopiero wpływ na ruch elementów mechanicznych robota. Ponadto w układach robotycznych często znaczenie mają bezwładności oraz poślizgi. Aby uwzględnić to wszystko w modelu konieczne jest wyprowadzenie dynamiki układu, będącej równaniami opisującymi odpowiedź układu na zadane sterowanie.

Model dynamiki może pozwolić na osiągnięcie znacznie lepszych rezultatów sterowania, czy jest to realizacja np. śledzenia trajektorii czy sterowanie do punktu. Nie uwzględniając dynamiki układu pomijany jest wpływ wielu sił występujących w rzeczywistym robocie, w tym przypadku istotne będą siły tarcia, jako że nie jest możliwy ruch robota po ścieżce innej niż prosta bez występowania poślizgów.

### 2.1 Zasada d'Alemberta

Niech  $q \in R^n$  będzie wektorem współrzędnych opisujących zachowanie robota spełniającym  $l$  niezależnych ograniczeń fazowych w postaci macierzy Pfaffa, której wiersze są pojedynczymi ograniczeniami

$$A(q)\dot{q} = 0. \quad (2.1)$$

Zgodnie z zasadą d'Alemberta siły więzów  $F$ , wymuszające spełnienie ograniczeń nieholonomicznych, nie wykonują pracy na dopuszczalnych trajektoriach układu [10]. Zachodzi zatem zależność

$$F^T \dot{q} = 0. \quad (2.2)$$

Z powyższych równań wynika, iż  $F^T$  jest kombinacją liniową kolumn macierzy Pfaffa  $A(q)$ , a zatem

$$F^T = \lambda^T A(q), \quad (2.3)$$

gdzie  $\lambda$  jest wektorem mnożników Lagrange'a.

## 2.2 Model we współrzędnych uogólnionych

Zdefiniujmy lagranżjan układu bez ograniczeń fazowych

$$L(q, \dot{q}) = K(q, \dot{q}) - V(q), \quad (2.4)$$

gdzie  $K(q, \dot{q}) = \frac{1}{2} \dot{q}^T P(q) \dot{q}$  — energia kinetyczna układu,  $V(q)$  — energia potencjalna układu. Równania ruchu układu z ograniczeniami możemy uzyskać korzystając z zasady d'Alemberta

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = A^T(q) \lambda. \quad (2.5)$$

Wprowadzając do układu siły zewnętrzne  $u$  otrzymamy postać

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = A^T(q) \lambda + B(q) u, \quad (2.6)$$

gdzie  $u$  — wektor uogólnionych sił zewnętrznych,  $B(q)$  — macierz wejściowa.

Wykorzystując funkcję Lagrange'a oraz postać ograniczeń możemy przekształcić równania dynamiki (2.6) do postaci

$$P(q) \ddot{q} + C(q, \dot{q}) \dot{q} + D(q) = A^T \lambda + B(q) u, \quad (2.7)$$

gdzie  $P$  — symetryczna, dodatnio określona macierz inercji,  $C$  — macierz sił odśrodkowych i Coriolisa,  $D$  — wektor sił potencjalnych (grawitacji).

Macierz sił Coriolisa możemy wyznaczyć wykorzystując symbole Cristoffela pierwszego rodzaju

$$\Gamma_{ijk}(q) = \frac{1}{2} \left( \frac{\partial P_{ij}}{\partial q_k} + \frac{\partial P_{ik}}{\partial q_j} - \frac{\partial P_{jk}}{\partial q_i} \right), \quad (2.8)$$

$$C_{ij}(q, \dot{q}) = \sum_k C_{jk}^i(q) \dot{q}_k. \quad (2.9)$$

Warto również uwzględnić w modelu siły tarcia  $F_f$  oraz reakcji pochodzące od podłoża  $F_s$ . Jako, że równanie (2.7) jest odpowiednikiem II Zasady Dynamiki Newtona, siły

$$F(q, \dot{q}) = F_f(q, \dot{q}) + F_s(q, \dot{q}),$$

możemy dołączyć po jego prawej stronie otrzymując

$$P(q) \ddot{q} + C(q, \dot{q}) \dot{q} + D(q) = A^T \lambda + F(q, \dot{q}) + B(q) u. \quad (2.10)$$

## 2.3 Model we współrzędnych pomocniczych

Opierając się na równaniu (2.2) wiemy, że dla układu istnieją pomocnicze prędkości, dla których zachodzi

$$\dot{q} = G(q)\eta, \quad (2.11)$$

gdzie macierz  $G$  spełnia warunek

$$A(q)G(q) = 0, \quad (2.12)$$

a  $\eta$  to wektor prędkości pomocniczych. Korzystając z równania (2.11) oraz faktu, że  $G^T(q)A^T(q) = 0$  możemy przekształcić równania ruchu z ograniczeniami fazowymi do formy bez mnożników Lagrange'a

$$P^*\dot{\eta} + C^*\eta + D^* = F^* + B^*u, \quad (2.13)$$

gdzie  $P^* = G^T P(\dot{G}\eta + G\dot{\eta})$ ,  $C^* = G^T C$ ,  $D^* = G^T D$ ,  $F^* = G^T F$ ,  $B^* = G^T B$ .

Równania ruchu układu (2.11, 2.13) możemy zapisać jako

$$\begin{cases} \dot{q} = G(q)\eta \\ \dot{\eta} = (G^T(q)P(q)G(q))^{-1}(-G^T(q)(P(q)\dot{G}(q)\eta) - C(q, \dot{q}) + F(q, \dot{q}) + B(q)u) \\ y = k(q) \end{cases}, \quad (2.14)$$

gdzie  $y=k(q)$  jest funkcją wyjścia układu określającą współrzędne platformy, których wartość będą śledzone. W układzie nie występuje macierz  $D$  jako, że przyjmujemy iż robot porusza się po płaskim podłożu i siły grawitacji nie mają wpływu na energie. Zauważmy, że jeżeli

$$\begin{aligned} \bar{X} &= (G^T(q)P(q)G(q))^{-1}G^T(q), \\ \bar{Y} &= (-P(q)\dot{G}(q)\eta - C(q, \dot{q}) + F(q, \dot{q})), \end{aligned}$$

równania (2.14) wyrazić w formie

$$\begin{pmatrix} \dot{\omega} \\ \dot{\eta} \end{pmatrix} = \begin{pmatrix} G\eta \\ \bar{X}\bar{Y} \end{pmatrix} + \begin{pmatrix} 0 \\ \bar{X}B \end{pmatrix} u$$

Wówczas zgodne będą z postacią

$$\dot{x} = f(x) + g(x)u = f(x) + \sum_{i=1}^m g_i(x)u_i, \quad y = k(x), \quad (2.15)$$

gdzie

$$x = (\omega, \dot{\omega}) = (x^1, x^2),$$

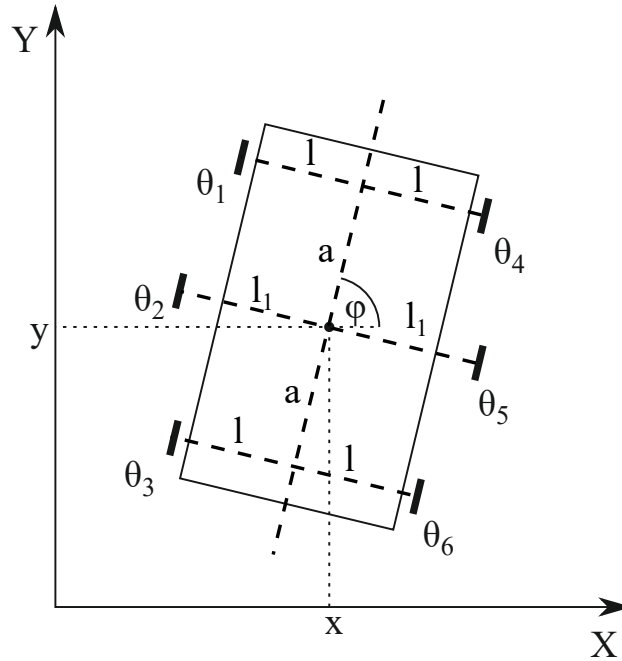
$$g(x) = [0_{5 \times 2}, P^{-1}(x^1)B],$$

$$f(x) = (x^2, P^{-1}(x^1)(-D(x) + F(x))).$$

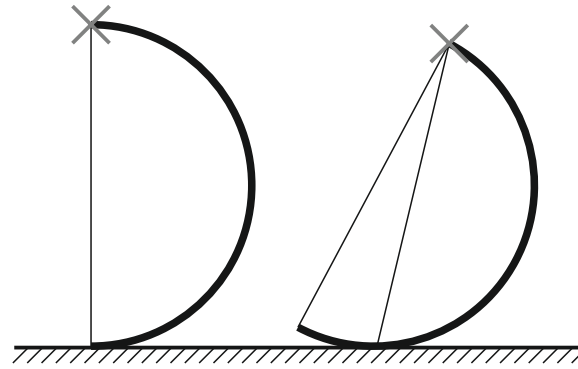
Niech  $T > 0$  oznacza horyzont czasu sterowania. Załóżmy, że dopuszczalne funkcje sterowania należą do przestrzeni Hilberta  $X = L_m^2[0, T]$  posiadającej iloczyn skalarny zdefiniowany jako  $\langle u_1(\cdot), u_2(\cdot) \rangle = \int_0^T u_1^T(t)u_2(t)dt$ .

Niech dla danego sterowania  $u(\cdot)$ ,  $x(t) = \phi_{x_0, t}(u(\cdot))$  będzie stanem trajektorii układu sterowania o warunkach początkowych  $x_0$ . Odpowiednia trajektoria wyjściowa ma postać  $y(t) = k(x(t))$ . Wówczas odwzorowanie (2.15) definiuje konfigurację robota w chwili  $T$ .

$$K_{x_0, T}(u(\cdot)) = k(x(T)) = k(\phi_{x_0, T}(u(\cdot))). \quad (2.16)$$



Rysunek 2.1 Schemat badanego robota.



Rysunek 2.2 Zależność punktu podparcia od pozycji nogi.

## 2.4 Model kinematyki

Uwzględnienie wszystkich aspektów ruchu robota w jego modelu jest niezwykle trudne. W związku z tym przyjęte zostały pewne uproszczenia. Rozpatrując przypadek chodu trójpodporowego robota z nogami typu whęg przyjmujemy, że przez połowę okresu chodu w fazie podparcia znajdują się nogi 1, 3 i 5, a przez drugą połowę 2, 4 i 6 (rys. 2.1). Taka konfiguracja zapewni nam stabilność chodu. Wówczas przez część okresu chodu możemy w sterowaniu przyjmować trójkołowego robota z dwoma kołami 1, 3 i 5, a przez drugą część — model z kołami 2, 4 i 6.

Dodatkowym przyjętym uproszczeniem jest założenie stałego promienia koła oraz sprzężenie ruchu przedniego koła z tylnym po lewej i prawej stronie robota. Kształt nogi badanego robota jest połową okręgu, a co za tym idzie zależnie od orientacji nogi zmieniać się będzie lokalizacja punktu kontaktu nogi z podłożem względem lokalnego układu współrzędnych robota (rys. 2.2).

Pełen wektor konfiguracji robota ma postać  $q = (x, y, \phi, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6)^T$ , gdzie  $x, y$  — położenie środka robota w globalnym układzie współrzędnych;  $\phi$  — orientacja korpusu robota w globalnym układzie współrzędnych;  $\theta_i$  — kąt obrotu  $i$ -tego koła. Jednakże,

jako że analizujemy sytuację w której w danej chwili robot ma kontakt z podłożem jedynie trzema kołami otrzymamy dwa prostsze wektory stanu dla dwóch modeli. Dodatkowo przyjmujemy, że po każdej ze stron robota ruch koła przedniego został sprzężony z ruchem koła tylnego. Wówczas wektory konfiguracji będą miały postać  $q_1 = (x, y, \phi, \theta_{13}, \theta_5)^T$  oraz  $q_2 = (x, y, \phi, \theta_2, \theta_{46})^T$ , gdzie  $\theta_{13}, \theta_{46}$  oznaczają kąty obrotu sprzężonych kół. Poniżej wyprowadzono równania modelu dla pierwszego wektora konfiguracji, drugi model uzyskujemy analogicznie do pierwszego.

Położenie kół (zobacz rys. 2.1) w globalnym układzie współrzędnych to

$$\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} x + \cos \phi \cdot a - \sin \phi \cdot l \\ y + \cos \phi \cdot l + \sin \phi \cdot a \end{pmatrix}, \quad (2.17)$$

$$\begin{pmatrix} x_3 \\ y_3 \end{pmatrix} = \begin{pmatrix} x - \cos \phi \cdot a - \sin \phi \cdot l \\ y + \cos \phi \cdot l - \sin \phi \cdot a \end{pmatrix}, \quad (2.18)$$

$$\begin{pmatrix} x_k \\ y_k \end{pmatrix} = \begin{pmatrix} x - \sin \phi \cdot l_1 \\ y - \cos \phi \cdot l_1 \end{pmatrix}, \quad (2.19)$$

gdzie  $x_k, y_k$  — położenie  $k$ -tego koła w globalnym układzie współrzędnych;  $a, l, l_1$  — odległości koła od środka robota względem lokalnych osi  $x$  i  $y$ .

Warunki na brak poślizgu wzdłużnego i poprzecznego kolejno kół 1, 3 i 5 będą miały wówczas postać

$$\begin{aligned} l \cdot \sin \phi + a \cdot \cos \phi - R \cdot \dot{\theta}_1 &= 0, \\ l \cdot \cos \phi - a \cdot \sin \phi &= 0, \end{aligned} \quad (2.20)$$

$$\begin{aligned} l \cdot \sin \phi - a \cdot \cos \phi - R \cdot \dot{\theta}_3 &= 0, \\ l \cdot \cos \phi + a \cdot \sin \phi &= 0, \end{aligned} \quad (2.21)$$

$$\begin{aligned} -l_1 \cdot \sin \phi - R \cdot \dot{\theta}_5 &= 0, \\ -l_1 \cdot \cos \phi &= 0, \end{aligned} \quad (2.22)$$

gdzie  $R$  to promień koła.

### 2.4.1 Model z dopuszczonymi poślizgami poprzecznymi

Macierz Pfaffa uwzględniająca wszystkie możliwe ograniczenia fazowe ma postać

$$A = \begin{bmatrix} \cos(\phi(t)) & \sin(\phi(t)) & -l & -R & 0 \\ \cos(\phi(t)) & \sin(\phi(t)) & l_1 & 0 & -R \\ -\sin(\phi(t)) & \cos(\phi(t)) & a & 0 & 0 \\ -\sin(\phi(t)) & \cos(\phi(t)) & 0 & 0 & 0 \\ -\sin(\phi(t)) & \cos(\phi(t)) & -a & 0 & 0 \end{bmatrix}. \quad (2.23)$$

Ze względu na jednakową prędkość dla kół znajdujących się na jednym boku robota przyjęto wspólny punkt niepodlegający poślizgom wzdłużnym znajdujący się pomiędzy tymi kołami. Aby możliwe było poruszanie się robota po krzywiźnie dopuszczone zostały poślizgi poprzeczne. Zatem nasza macierz Pfaffa będzie postaci

$$A = \begin{bmatrix} \cos(\phi(t)) & \sin(\phi(t)) & -l & -R & 0 \\ \cos(\phi(t)) & \sin(\phi(t)) & l_1 & 0 & -R \end{bmatrix} \quad (2.24)$$

Macierz

$$\bar{A} = \begin{bmatrix} -\sin(\phi(t)) & \cos(\phi(t)) & a & 0 & 0 \\ -\sin(\phi(t)) & \cos(\phi(t)) & 0 & 0 & 0 \\ -\sin(\phi(t)) & \cos(\phi(t)) & -a & 0 & 0 \end{bmatrix}. \quad (2.25)$$

zawiera ograniczenia nieuwzględnione w kinematyce.

Następnie model przekształcono do postaci bezdryfowego układu sterowania

$$\dot{q} = G(q) \cdot \eta. \quad (2.26)$$

Wykorzystując własność (2.12) wyznaczono macierz

$$G = \begin{bmatrix} \sin(\phi(t)) & 0 & R \cos(\phi(t)) \\ -\cos(\phi(t)) & 0 & R \sin(\phi(t)) \\ 0 & R & 0 \\ 0 & l & 1 \\ 0 & -l_1 & 1 \end{bmatrix}. \quad (2.27)$$

## 2.5 Model dynamiki robota trójkąowego

Energia kinetyczna platformy wynosi

$$E_{kp} = \frac{1}{2}I_z\dot{\phi}^2 + \frac{1}{2}M_p\dot{x}^2\dot{y}^2, \quad (2.28)$$

gdzie  $I_z$  — moment bezwładności platformy względem osi Z,  $M_p$  — masa platformy. Energia kinetyczna kół to

$$E_{kk1} = \frac{1}{2}(I_{xx}\dot{\theta}_1^2 + I_{zz}\dot{\phi}^2 + M_k(\dot{x}^2 + \dot{y}^2 + l^2\dot{\phi}^2 + a^2\dot{\phi}^2 - 2l\dot{\phi}(\dot{y}\sin\phi + \dot{x}\cos\phi) + 2a\dot{\phi}(\dot{y}\cos\phi - \dot{x}\sin\phi))), \quad (2.29)$$

$$E_{kk3} = \frac{1}{2}(I_{xx}\dot{\theta}_3^2 + I_{zz}\dot{\phi}^2 + M_k(\dot{x}^2 + \dot{y}^2 + l^2\dot{\phi}^2 - a^2\dot{\phi}^2 - 2l\dot{\phi}(\dot{y}\sin\phi + \dot{x}\cos\phi) - 2a\dot{\phi}(\dot{y}\cos\phi - \dot{x}\sin\phi))), \quad (2.30)$$

$$E_{kk5} = \frac{1}{2}(I_{xx}\dot{\theta}_5^2 + I_{zz}\dot{\phi}^2 + M_k(\dot{x}^2 + \dot{y}^2 - l_1^2\dot{\phi}^2 + 2l_1\dot{\phi}(\dot{y}\sin\phi + \dot{x}\cos\phi))), \quad (2.31)$$

gdzie  $M_p$  — masa platformy,  $M_k$  — masa koła,  $I_{xx}$  — moment bezwładności koła względem lokalnej osi X,  $I_{zz}$  — moment bezwładności koła względem lokalnej osi Z. Po zsumowaniu wszystkich energii kinetycznych wyznaczono macierz bezwładności

$$P = \begin{bmatrix} P_{11} & 0 & P_{13} & 0 & 0 \\ 0 & P_{22} & P_{23} & 0 & 0 \\ P_{31} & P_{32} & P_{33} & 0 & 0 \\ 0 & 0 & 0 & P_{44} & 0 \\ 0 & 0 & 0 & 0 & P_{55} \end{bmatrix}, \quad (2.32)$$

gdzie

$$\begin{aligned}
 P_{11} &= P_{22} = \frac{1}{2}(3M_k + M_p), \\
 P_{13} &= P_{31} = -M_k(2l - l_1) \cos \phi, \\
 P_{23} &= P_{32} = -M_k(2l - l_1) \sin \phi, \\
 P_{33} &= \frac{1}{2}((I_z + 3I_{zz} + (2a^2 + 2l^2 + l_1^2)M_k), \\
 P_{44} &= I_{xx}, \\
 P_{55} &= \frac{I_{xx}}{2}.
 \end{aligned}$$

Macierz sił odśrodkowych oraz Coriolisa wyznaczona została z wykorzystaniem symboli Cristoffela pierwszego rodzaju

$$C = \begin{bmatrix} 0 & 0 & -(2l - l_1) \sin(\phi(t)) & 0 & 0 \\ 0 & 0 & M_k(-2l + l_1) \cos(\phi(t)) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (2.33)$$

W modelu uwzględniono dwa rodzaje sił tarcia. Pierwszą z nich są ogólne siły tarcia oddziałujące liniowo na współrzędne stanu robota

$$F_t(q, \dot{q}) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\epsilon_1 \theta'_{13}(t) \\ -\epsilon_2 \theta'_5(t) \end{pmatrix}. \quad (2.34)$$

Drugim rodzajem sił tarcia są siły reakcji podłoża odpowiadające poślizgom poprzecznym

$$\begin{aligned}
 F_s(q, \dot{q}) &= \bar{A}^T(q)r, \\
 r &= -\beta \bar{A},
 \end{aligned}$$

gdzie  $\beta$  — diagonalna macierz, której elementy na przekątnej  $\beta_i = N_i \cdot \epsilon_i$ ;  $N_i$  — siły nacisku kół na podłoże;  $\epsilon_i$  — współczynniki tarcia. Otrzymamy wówczas wektor sił reakcji

$$F_s(q, \dot{q}) = \begin{pmatrix} -\epsilon_3(N_1 + N_2) \sin(\phi(t)) (\sin(\phi(t))x'(t) - \cos(\phi(t))y'(t)) \\ \epsilon_4(N_1 + N_2) \cos(\phi(t)) (\sin(\phi(t))x'(t) - \cos(\phi(t))y'(t)) \\ 0 \\ 0 \\ 0 \end{pmatrix}. \quad (2.35)$$

W zadaniu sterowania interesować nas będzie śledzenie współrzędnych  $x$ ,  $y$ ,  $\phi$  platformy, stąd

$$y^* = k(x^*) = \begin{pmatrix} x \\ y \\ \phi \end{pmatrix}. \quad (2.36)$$

## 2.6 Parametry badanej konstrukcji

Wszystkie badania przeprowadzono przyjmując następujące parametry:  $M_p = 3.2kg$  — masa platformy,  $M_k = 0.025kg$  — masa koła,  $L = 0.35m$  — długość platformy,  $W = 0.25m$  — szerokość platformy,  $W_k = 0.014m$  — szerokość koła,  $I_z = \frac{1}{12}M_p(L^2 + W^2)$ ,  $I_{zz} = \frac{1}{12}M_k(3R^2W_k^2)$ ,  $I_{xx} = \frac{1}{2}M_kR^2$ ,  $l = 0.1m$ ,  $l_1 = 0.15m$ ,  $a = 0.15m$ .



# Rozdział 3

## Algorytmy sterowania robota

Projektowanie algorytmu sterowania dla robota z nogami typu wheg uwzględniającego wszystkie niuanse jego konstrukcji byłoby niezwykle trudnym zadaniem. Dlatego obowiązujące są założenia przyjęte we wstępie.

Rozpatrywany będzie jedynie przypadek chodu trójpodporowego. Nogami znajdującymi się w danej chwili w fazie podparcia będą nogi 1, 3 i 5 lub 2, 4 i 6. Nogami znajdującymi się w fazie przenoszenia nie będą uwzględniane, jako że nie mają bezpośrednio znacznego wpływu na sposób poruszania się robota. Dodatkowo nogi znajdujące się w fazie podparcia będą rozpatrywane jako koła. Może to prowadzić do pewnych rozbieżności względem rzeczywistego robota z dwóch względów. Pierwszym z nich jest inna bezwładność. Nie powinno to być jednak zbyt znaczące, gdyż w rzeczywistym robocie większość masy nogi skupia się w metalowym hubie mocującym, który jest w przybliżeniu walcem o jednorodnie rozłożonej masie. Drugą i znacznie bardziej znaczącą różnicą jest zmienny punkt kontaktu nogi z podłożem względem robota, który nie będzie uwzględniany w przypadku koła.

### 3.1 Prosty algorytm ruchu po krzywiźnie

Najprostszym podejściem do sterowania robotem jest kontrola jego ruchu po krzywej o zadanej krzywiźnie. Wówczas zadaniem jest wyznaczenie prędkości dla każdego z kół robota znajdujących się w fazie podparcia tak, aby odpowiadały spodziewanej prędkości liniowej w punkcie mocowania koła przy założonym promieniu krzywizny i prędkości ruchu środka robota [11]. Nie jest możliwe jednak dokładne zrealizowanie oczekiwanej prędkości ze względu na niemożność obrotu nóg wzdłuż osi  $Z$ . Konieczne jest zatem przyjęcie dopuszczenia pełnych poślizgów poprzecznych i realizowanie jedynie składowej prędkości prostopadłej do osi obrotu nogi.

Powyższe założenia pozwalają na wyprowadzenie prędkości liniowej dowolnej z nóg w fazie podparcia

$$V_{yn} = \frac{V_{sr} \cdot \sqrt{(R + d_{xn})^2 + d_{yn}^2}}{R \cdot \sqrt{\left(\frac{d_{yn}}{R + d_{xn}}\right)^2 + 1}}, \quad (3.1)$$

gdzie  $V_{sr}$  — prędkość liniowa środka robota,  $R$  — promień krzywizny,  $d_{xn}$  — przesunięcie  $n$ -tego koła względem środka robota wzdłuż lokalnej osi  $X$ ,  $d_{yn}$  — przesunięcie  $n$ -tego koła względem środka robota wzdłuż lokalnej osi.

Aby możliwe było zrealizowanie chodu konieczne jest również sterowanie prędkością ruchu nóg znajdujących się w fazie przenoszenia. Ich prędkość liniowa musi być dobrana

w taki sposób, aby w momencie rozpoczynania swojej fazy podparcia znajdowały się w odpowiednim miejscu. Własności takiego sposobu sterowania wraz z symulacjami i eksperymentami zawarto w pracy [11].

## 3.2 Pomocniczy algorytm sterowania do punktu

Niech zadaniem sterowania dla układu (2.15) będzie osiągnięcie zadanego położenia  $y_d$  w przestrzeni zadaniowej przy warunkach początkowych  $x_0$  w czasie  $T$ . Należy znaleźć takie sterowanie  $u_d(t)$ , aby odpowiedź systemu znajdującego się w położeniu początkowym  $y_0 = k(x_0)$  osiągnęła docelowy punkt  $K_{x_0,T}(u_d(\cdot)) = y_d$ .

Poniżej rozważamy sformułowanie zadania dla robota kołowego z poślizgami. Zastosowanie tego rozwiązania dla robota trójkołowego opisanego w podrozdziałach 2.4 i 2.5 pozwoli w efekcie na rozwiązanie problemu dla robota z nogami typu wheg, co opisano w podrozdziale 3.3.

Zadanie planowania ruchu może być rozwiązane z wykorzystaniem algorytmu opartego na Jakobianie wywodzącego się z podejścia endogenicznej przestrzeni konfiguracyjnej (ECSA) [8].

Zasada działania algorytmu jest następująca. Rozpoczynamy z dowolną funkcją sterowania  $u_0(\cdot) \in X$ . Jeżeli ta funkcja rozwiązuje zadanie algorytm kończy swoje działanie. W przeciwnym wypadku szukamy różniczkowalnej krzywej  $u_\theta(\cdot)$  przechodzącej przez  $u_0$  takiej, że błąd położenia w przestrzeni stanów  $e(\theta) = K_{x_0,T}(u_\theta(\cdot)) - y_d$  maleje eksponencjalnie wraz z krzywą  $u_\theta$  z określoną prędkością zaniku błędu  $\gamma > 0$ . Po zróżniczkowaniu błędu względem  $\theta$  otrzymamy

$$DK_{x_0,T}(u_\theta(\cdot)) \frac{du_\theta(\cdot)}{d\theta} = J_{x_0,T}(u_\theta(\cdot)) \frac{du_\theta(\cdot)}{d\theta} = -\gamma e(\theta). \quad (3.2)$$

Wyrażenie

$$J_{x_0,T}(u_\theta(\cdot))v(\cdot) = \rho(T) = C(T) \int_0^T \Phi(T,s)B(s)v(s)ds \quad (3.3)$$

jest Jakobianem systemu dla endogenicznej konfiguracji  $u(\cdot)$ . Dla zadanego  $v(\cdot) \in X$  Jakobian określa wartości wyjściowe w chwili  $T$  liniowej aproksymacji

$$\dot{\xi}(t) = A(t)\xi(t) + B(t)v(t), \rho(t) = C(t)\xi(t), \xi(0) = 0, \quad (3.4)$$

systemu (2.15) z macierzami

$$\begin{aligned} A(t) &= \frac{\partial(f(x(t)) + g(x(t))u(t))}{\partial x}, \\ B(t) &= g(x(t)), \\ C(t) &= \frac{\partial k(x(t))}{\partial x}. \end{aligned}$$

Macierz przejścia  $\Phi(t,s)$  jest rozwiązaniem równania

$$\frac{\partial \Phi(t,x)}{\partial t} = A(t)\Phi(t,s),$$

gdzie  $\Phi(s,s) = \mathbb{I}_{10}$ . Wstawiając do równania (3.2) odwrotność Jakobianu (3.3), czyli  $J_{x_0,T}^\#(u_\theta(\cdot))$ , otrzymamy algorytm dający rozwiązanie zadania planowania ruchu jako granicę  $u_d(t) = \lim_{\theta \rightarrow +\infty} u_\theta(t)$  rozwiązania równania różniczkowego

$$\frac{du(\cdot)}{d\theta} = -\gamma J_{x_0,T}^\#(u_\theta(\cdot))(K_{x_0,T}(u_\theta(\cdot)) - y_d). \quad (3.5)$$

Używając pseudoodwrotności Moore'a–Penrose'a otrzymamy równanie

$$\frac{du(t)}{d\theta} = -\gamma B_\theta^T(t) \Phi_\theta^T(T, t) C_\theta^T(T) G_{x_0,T}^{-1}(u(\cdot)) \cdot (K_{x_t,T}(u(\cdot)) - y_d), \quad (3.6)$$

gdzie indeks dolny  $\theta$  oznacza, że wartość ma być wyliczona z użyciem  $(u_\theta(t), x_\theta(t))$ .  $G_{x_0,T}$  jest macierzą Grama, którą wyznaczyć można całkując równania różniczkowe Lapunowa

$$\dot{M}(t) = B(t)B^T(t) + A(t)M(t) + M(t)A^T(t), \quad (3.7)$$

z  $M(0) = 0$  i przyjętym  $G_{x_0,T}(u(\cdot)) = C(T)M(T)C^T(T)$ .

Rozwiązanie równania (3.6) można uzyskać korzystając z aproksymacji szeregiem Fouriera. Niech  $u_\lambda(t) = P_s(t)\lambda$ , z  $P_s(t) = \text{diag}\{P(t), P(t)\}$  oznaczającym blokowo diagonalną macierz w elementami  $P(t) = [1, \sin\omega t, \cos\omega t, \dots, \cos p\omega t]$ ,  $\omega = 2\pi/T$ . Wykorzystując tę parametryzację Jakobian przyjmuje postać macierzową

$$J_{x_0,T}(\lambda) = C_\lambda(T) \int_0^T \Phi_\lambda(T, t) B_\lambda(t) P_s(t) dt, \quad (3.8)$$

zatem Jakobian  $J_{x_0,T}(\lambda) = C_\lambda(T)J_\lambda(T)$ , gdzie  $J_\lambda$  rozwiązuje równanie

$$\dot{J}_\lambda(t) = A_\lambda(t)J_\lambda(t) + B_\lambda(t)P_s(t). \quad (3.9)$$

przy warunku początkowym  $J_\lambda(0) = 0$ . Po zastosowaniu metody Eulera otrzymamy parametry sterowania w formie

$$\lambda_{\theta+1} = \lambda_\theta - \gamma J_{x_0,T}^T(\lambda_\theta) G_{x_0,T}^{-1}(K_{x_0,T}(\lambda_\theta) - y_d), \quad (3.10)$$

$$\theta = 0, 1, \dots,$$

$$G_{x_0,T}^{-1} = J_{x_0,T}(\lambda) J_{x_0,T}^T(\lambda).$$

Rozwiązaniem zadania planowania ruchu będą sterowania

$$u_d(t) = P_s(t)\lambda_d, \quad (3.11)$$

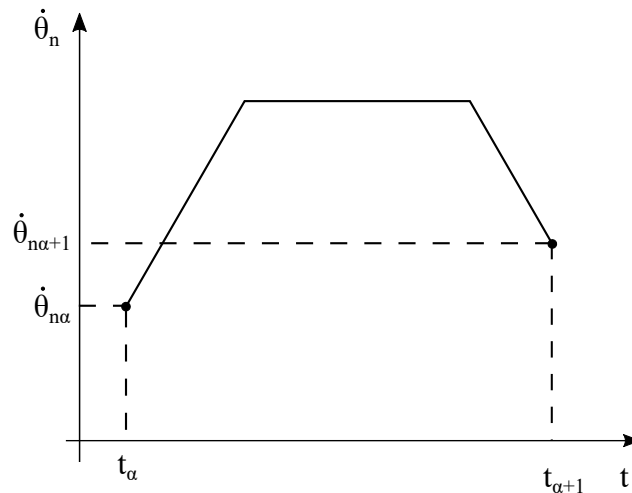
$$\lambda_d = \lim_{\theta \rightarrow +\infty} \lambda_\theta.$$

### 3.3 Pełny algorytm sterowania do punktu

W podrozdziale 3.2 opisano algorytm sterowania do punktu dla robota kołowego z poślizgami. Stanowi on bazę dla pełnego algorytmu. Poniżej sformułowano algorytm sterowania do punktu robota z nogami typu whęg opierający się na algorytmie wspomnianym powyżej.

Rozwiązanie sformułowanego w podrozdziale 3.2 zadania sterowania dla robota z nogami typu whęg polega na naprzemiennym rozwiązywaniu tego zadania dla robota trójkołowego z dwoma kołami umieszczonymi raz po jednej, a raz po drugiej stronie robota. Umiejętne przełączanie modeli dla rozwiązań częściowych pozwoli na osiągnięcie zadanego celu.

Dla ustalenia nomenklatury model z kołami 1, 3 i 5 nazwijmy modelem "A", zaś z kołami 2, 4 i 6 — modelem "B". Przez  $\theta_p$  oznaczmy wartość położenia kąтового nogi w



Rysunek 3.1 Profil prędkości ruchu nogi podczas fazy przenoszenia.

której rozpoczyna się faza podparcia, a przez  $\theta_k$  wartość dla której się ona kończy. Wartości te przyjmowane są arbitralnie i decydują o długości pojedynczego kroku. Załóżmy, że w konfiguracji początkowej  $q_0$  robot stoi na nogach 1, 3 i 5 tak, że  $\theta_{13} = \theta_5 = \theta_p$ . Przyjmijmy prędkość początkową robota  $\dot{q}_0$  i określmy punkt docelowy  $y_d$  oraz horyzont czasu  $T$ . W pierwszym kroku algorytmu rozwiążmy zadanie sterowania dla modelu "A" dla zadanego  $q_0$ ,  $y_d$  i  $T$  stosując metodę przestrzeni endogenicznej (3.10, 3.11). Stosujemy otrzymane rozwiązanie do chwili  $t_{\alpha 1}$ , w której dowolna z z nóg 1, 3 i 5 osiągnie położenie  $\theta_k$  jako pierwsza. Zapamiętujemy konfigurację robota w tej chwili jako  $q_{t_\alpha} = (x(t_\alpha), y(t_\alpha), \phi(t_\alpha), \theta_{13}(t_\alpha), \theta_5(t_\alpha))^T$  oraz jego prędkości  $\dot{q}_{t_\alpha}$ . Zapamiętujemy też sterowania  $u_i(t)$ ,  $0 < t < t_\alpha$ , gdzie  $i$  — numer kroku, tutaj 1.

W kolejnym kroku powtarzamy czynności wykonane w kroku poprzednim wykorzystując model "B" przy warunkach początkowych  $q_{t_\alpha}$ ,  $\dot{q}_{t_\alpha}$ , zadanym  $y_d$  i horyzoncie czasu  $T$ . Położenie początkowe nóg 2, 4 i 6 przyjmujemy jako  $\theta_2 = \theta_{46} = \theta_p$  zaś ich prędkości wyznaczamy tak, aby w momencie wejścia nóg w kontakt z podłożem odpowiadały one prędkościom liniowym punktów, w których są one zamocowane. W tym celu możemy skorzystać ze wzoru (3.1). Z tak dobranymi warunkami początkowymi możliwe jest ponowne użycie metody endogenicznej pamiętając, że ruch rozpoczynamy teraz w chwili  $t_{\alpha i}$ , w tym wypadku  $i = 1$ . Powyższe działania należy powtarzać naprzemiennie dla modeli "A" i "B" do momentu osiągnięcia czasu  $T$ , w którym robot powinien osiągnąć punkt  $y_d$ .

Zagadnieniem, które należy jeszcze określić jest kwestia synchronizacji nóg i sposób ich sterowania w fazie przenoszenia. Podstawowym zadaniem jest tutaj realizacja takiego sterowania, aby

W chwili przełączania między modelami znany jest czas trwania ostatniego kroku. Na podstawie położenia nogi w chwili zakończenia fazy podparcia, warunków początkowych kolejnego kroku oraz czasu trwania kroku możliwe jest wyznaczenie takiego profilu prędkości dla nogi, aby spełnione zostały założenia warunków początkowych. Możliwa jest realizacja tego zadania poprzez jednostajne rozpędzanie nogi do maksymalnej prędkości a następnie jej spowalnianie do osiągnięcia pożądanego prędkości. Należy pamiętać, że pole powierzchni pod wykresem 3.1 wynosić będzie  $\phi_0 + 2\pi - \phi_k$ , gdzie  $\phi_0 + 2\pi$  odpowiada pozycji nogi rozpoczynającej fazę podparcia w kolejnym kroku.

# Rozdział 4

## Badania symulacyjne

Do wszystkich badań symulacyjnych wykorzystano oprogramowanie Wolfram Mathematica. Środowisko to uznano za odpowiednie do wykonanych prac ze względu na rozbudowane możliwości obliczeń symbolicznych i numerycznych.

### 4.1 Model robota trójkołowego

Przeprowadzone zostały eksperymenty z wykorzystaniem modelu trójkołowego robota z uwzględnieniem różnych parametrów dla sił reakcji poślizgów bocznych.

#### 4.1.1 Weryfikacja modelu poślizgów bocznych

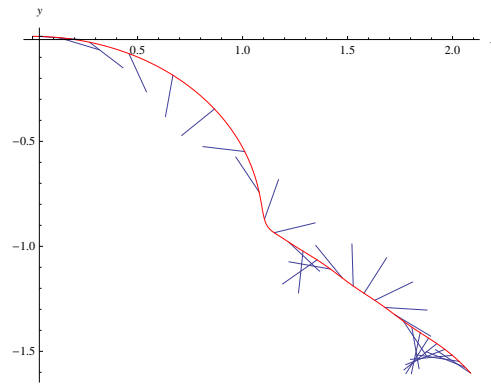
Badania symulacyjne rozpoczęto od weryfikacji modelu robota z poślizgami bocznymi. Na model w konfiguracji początkowej  $q_0 = 0, \dot{q}_0 = 0$  zadano sterowania  $u_1 = 0.006, u_2 = 0.005$  przez 10 sekund o obserwowano jego reakcję przez czas  $t_{max} = 30s$ . Oczekiwana odpowiedź rzeczywistego robota byłby ruch po krzywiźnie o dużym promieniu. W przypadku braku uwzględnienia sił reakcji poślizgów poprzecznych (jedyne tarcia uwzględnione w modelu to siły oporu toczenia kół) wynikiem jest robot swobodnie obracający się wokół własnej osi (rys. 4.1).

Dołączenie do modelu sił reakcji podłoża znacząco wpłynęło na otrzymane rezultaty. Przy niskich współczynnikach tarcia robot ponownie poruszał się w znacznym poślizgu bocznym, jednak po zaniku sterowań zaobserwowano, że zorientował się on kołami zgodnie z kierunkiem ruchu (rys. 4.2). Dalsze zwiększanie współczynników tarcia skutkowało zmniejszaniem występujących poślizgów poprzecznych i zwiększaniem promienia krzywej, po której poruszał się robot (rys. 4.3).

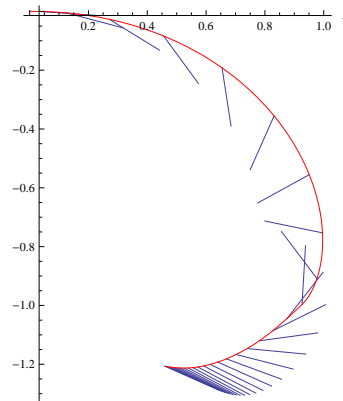
### 4.2 Sterowanie do punktu

W zadaniu sterowania do punktu robota trójkołowego przyjęto podobnie jak w poprzednim przypadku zerowe warunki początkowe zarówno dla położenia jak i prędkości robota. Horyzont czasu ustalono na  $T = 10s$ , a punkt zadany  $y_d = (2, 3, 1)$ . Wynik działania algorytmu dla modelu pozbawionego sił reakcji poślizgów poprzecznych przedstawiony jest na rysunku 4.4. Robot osiągnął zadany punkt z dokładnością rzędu  $10^{-14}$ , a zatem błąd sterowania jest na poziomie błędów numerycznych.

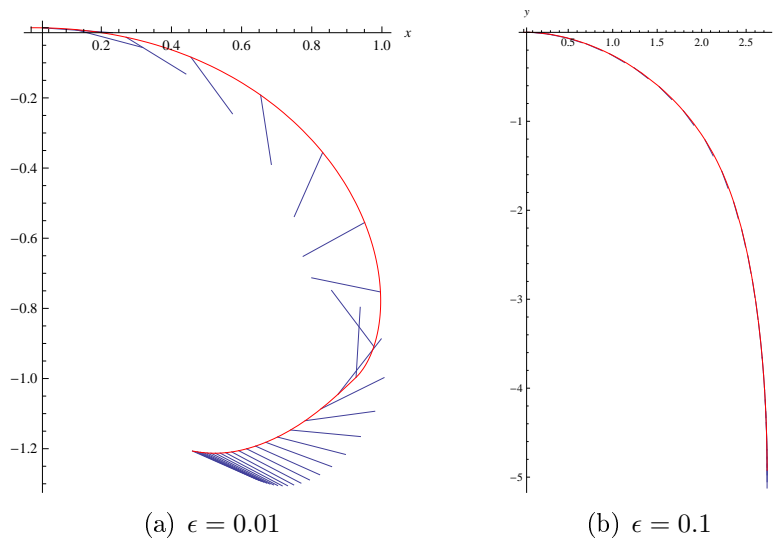
Kolejnym eksperymentem było powtórzenie symulacji na modelu uwzględniającym siły reakcji poślizgów poprzecznych. Ponownie wyniki sterowania spełniały oczekiwania. W



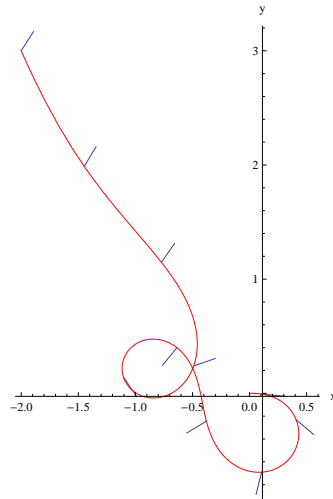
Rysunek 4.1 Ścieżka pokonana przez robota. Sterowania  $u_1 = 0.006, u_2 = 0.005$  przez 10s.  $t = 30s$ . Brak sił reakcji poślizgów poprzecznych. Czerwona linia oznacza ścieżkę, po której poruszał się robot, niebieskie odcinki oznaczają orientację robota.



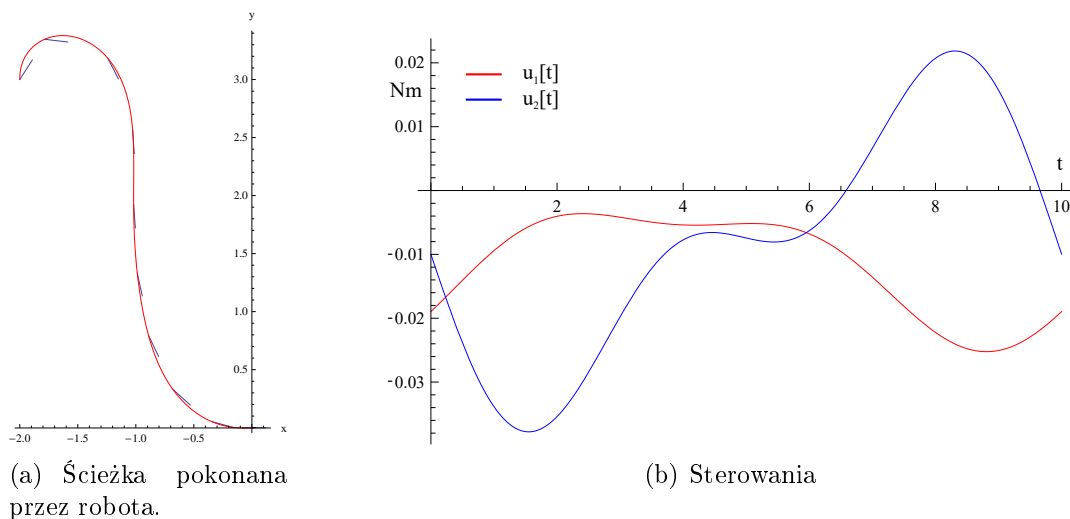
Rysunek 4.2 Ścieżka pokonana przez robota. Sterowania  $u_1 = 0.006, u_2 = 0.005$  przez 10s.  $t = 30s$ . Współczynnik tarcia poprzecznego  $\epsilon = 0.01$ .



Rysunek 4.3 Ścieżka pokonana przez robota. Sterowania  $u_1 = 0.006, u_2 = 0.005$  przez 10s.  $t = 30s$ .



Rysunek 4.4 Ścieżka pokonana przez robota sterowanego z zastosowaniem metody przestrzeni endogenicznej. Brak sił reakcji poślizgów poprzecznych.



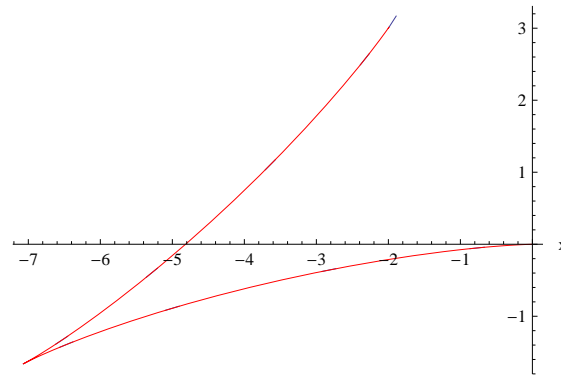
Rysunek 4.5 Wynik działania metody endogenicznej,  $\epsilon = 0.1$ .

otrzymanej trajektorii zauważalny jest wpływ sił tarcia pochodzących z poślizgów poprzecznych. Warto też zwrócić uwagę na otrzymane wartości sterowań, które są rozsądnie niskie, co jest istotne przy przenoszeniu sterowań na rzeczywistego robota.

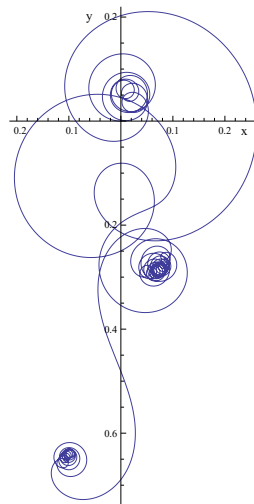
Wynik symulacji dla modelu o zadanych bardzo dużych współczynnikach tarcia pozwala na ciekawą obserwację. Ścieżka pokonana przez robota przywodzi bowiem na myśl sposób, w jaki człowiek manewrowałby pojazdem o bardzo dużym promieniu skrętu 4.6.

Metoda endogenicznej przestrzeni konfiguracyjnej bywa czuła na niekorzystnie sformułowane warunki początkowe, w szczególności początkowy wektor  $\lambda$ . Przykład wyniku działania algorytmu dla takiej sytuacji przedstawia rys. 4.7. Możliwe jest, że przy dopuszczeniu większej liczby iteracji algorytmu osiągnięty zostałyby zadowalające błędy sterowania, jednak długość pokonanej ścieżki dyskwalifikuje ten wynik jako użyteczny.

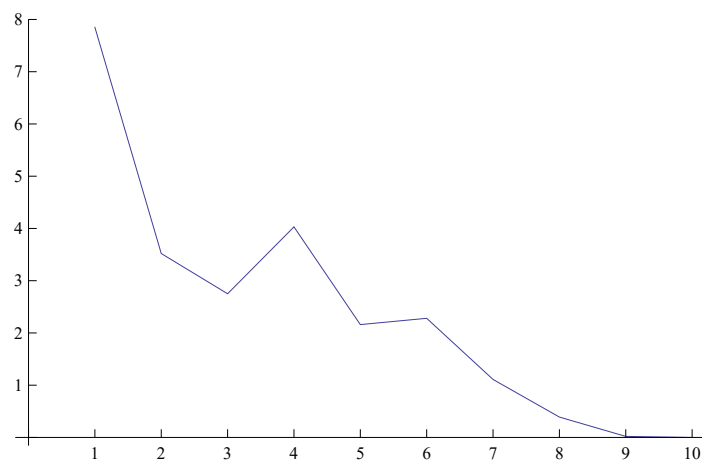
Warto zwrócić uwagę na zbieżność algorytmu sterowania. Wartość normy błędów sterowania w kolejnych iteracjach przedstawiona została na wykresie 4.8. W większości przypadków przy dobrze uwarunkowanych wartościach początkowych ok. 10 iteracji wystarcza, aby wyznaczyć zadowalające sterowania, a często wartość błędów osiąga małe wartości znacznie szybciej.



Rysunek 4.6 Wynik działania metody endogenicznej przy bardzo dużym współczynniku tarcia poprzecznego ( $\epsilon = 100$ ).



Rysunek 4.7 Wynik działania metody endogenicznej przy niekorzystnie uwarunkowanym wektorze  $\lambda_0$ .



Rysunek 4.8 Przykładowe wartości normy błędów sterowania w kolejnych iteracjach działania metody endogenicznej.



# Rozdział 5

## Podsumowanie

Celem pracy było zaprojektowanie algorytmu sterowania dla robota z nogami typu wheg uwzględniającego poślizgi. Został zaproponowany taki algorytm przy przyjęciu pewnych założeń — wykorzystaniu modelu uproszczonego w postaci trójkołowego robota o nieskrętnych kołach oraz sprzężeniu kół przednich i tylnych znajdujących się po tej samej stronie robota. Na tej podstawie wyprowadzono model robota z uwzględnieniem sił reakcji poślizgów poprzecznych. Dopuszczenie występowania poślizgów poprzecznych ze względu na budowę robota było konieczne, aby umożliwić ruch po krzywiźnie.

Wykorzystana w pracy metoda endogenicznej przestrzeni konfiguracyjnej wydaje się być odpowiednim narzędziem do realizacji sterowania badanym robotem. Zastosowana w sterowaniu modelem pomocniczym dawała bardzo dobre rezultaty. Czasem otrzymane sterowania nie zapewniały satysfakcjonujących wyników, jednak wystarczała wówczas modyfikacja początkowych wartości wektora  $\lambda$ . Zbieżność algorytmu również była zadowalająca, często dając dobre wyniki już w czwartej iteracji działania algorytmu. W efekcie można uznać ją jako stosunkowo wydajną. Nie byłoby możliwe implementowanie jej w niewielkim systemie wbudowanym, jednak uruchamiana na komputerze zazwyczaj dostarczała rozwiązanie w kilkadziesiąt sekund. Ponadto otrzymywane wartości sterowań utrzymywały się na rozsądnym poziomie, pozwalającym na zrealizowanie ich na fizycznym robocie. W pracy zaproponowano algorytm sterowania oparty na metodzie endogenicznej przestrzeni konfiguracyjnej połączony z przełączaniem modeli zależnie od fazy chodu.

W przyszłości warto by przeprowadzić identyfikację parametrów tarcia fizycznego robota poprzez przeniesienie na niego fizycznych sterowań i zestawienie wyników symulacji z rzeczywistymi. Byłoby to wstępem do implementacji algorytmu sterowania dla rzeczywistego robota. Ponadto na bazie obecnego modelu rozważyć można uwzględnienie kształtu nogi robota. Wiązałoby się to ze skomplikowaniem modelu, mogłoby jednak pozwolić na otrzymanie lepszych wyników sterowania.

Jako że napęd typu wheg zaprojektowany został z myślą o zachowaniu mobilności w trudnym terenie w przyszłości interesującym zadaniem może być zaprojektowanie algorytmu monitorującego stan robota w ruchu oraz występujące poślizgi. Model sił reakcji ulegałby wówczas ciągłym zmianom zależnie od podłoża po jakim aktualnie porusza się robot. Algorytm taki mógłby na bieżąco uwzględniać zmieniające się siły reakcji podłoża uzyskując prawdopodobnie bardzo dobre wyniki nawet na niejednorodnym podłożu.



# Bibliografia

- [1] Richard Altendorfer, Ned Moore, Haldun Komsuoglu, Martin Buehler, H Benjamin Brown Jr, Dave McMordie, Uluc Saranli, R Full, and Daniel E Koditschek.  
Rhex: a biologically inspired hexapod runner.  
*Autonomous Robots*, 2001.
- [2] Ivan Siles and Ian D Walker.  
Design, construction, and testing of a new class of mobile robots for cave exploration.  
In *Mechatronics, 2009. ICM 2009. IEEE International Conference on*, pages 1–6.  
IEEE, 2009.
- [3] Michał Burdka, Bogna Czyżewska, Paweł Joniak, Marek Malec, Bartosz Witkowski.  
Człapacz. projekt napędu kroczącoobrotowego.  
Projekt zespołowy. Politechnika Wrocławska, 2014.
- [4] Daniel E. Koditschek, Robert J Full, Martin Buehler.  
Mechanical aspects of legged locomotion control.  
*Arthropod structure & development*, 33(3):251–272, 2004.
- [5] Richard Harkins, Jason Ward, Ravi Vaidyanathan, AX Boxerbaum, and Roger D Quinn.  
Design of an autonomous amphibious robot for surf zone operations: part ii-hardware, control implementation and simulation.  
In *Advanced Intelligent Mechatronics. Proceedings, 2005 IEEE/ASME International Conference on*, pages 1465–1470. IEEE, 2005.
- [6] Uluc Saranli, Martin Buehler, Daniel E. Koditschek.  
Rhex: A simple and highly mobile hexapod robot.  
*The International Journal of Robotics Research*, 20(7):616–631, 2001.
- [7] Marcin Piątek.  
Problemy sterowania robotami kroczącymi - generatory chodu hexapoda.  
Praca doktorska. Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie, 2012.
- [8] Krzysztof Tchoń, Katarzyna Zadarnowska, Łukasz Juskiewicz, Krzysztof Arent.  
Modeling and control of a skid-steering mobile platform with coupled side wheels.  
*Bulletin of the Polish Academy of Sciences Technical Sciences*, 2015.
- [9] Krzysztof Kozłowski, Dariusz Pazderski.  
Modeling and control of a 4-wheel skid-steering mobile robot.  
*International journal of applied mathematics and computer science*, 14:477–496, 2004.
- [10] Krzysztof Tchoń, Alicja Mazur, Ignacy Dulęba, Robert Hossa, Robert Muszyński.  
*Manipulatory i roboty mobilne: modele, planowanie ruchu, sterowanie*.  
Problemy Współczesnej Nauki, Teoria i Zastosowania. Robotyka. Akademicka Oficyna Wydawnicza PLJ, 2000.

- [11] Bartosz Witkowski.  
Sterowanie sześcionożnego robota krocącego z nogami typu wheg.  
Praca inżynierska. Politechnika Wroclawska, 2015.