

Spis treści

1 Wstęp	2
1.1 Cel pracy	2
1.2 Zawartość pracy	2
2 Model manipulatora mobilnego	2
2.1 Model we współrzędnych uogólnionych	3
2.1.1 Ograniczenia nieholonomiczne	3
2.2 Model w prędkościach pomocniczych	4
3 Odsprzęganie wejściowo-wyjściowe dla manipulatora mobilnego	5
3.1 Podstawowy algorytm Yamamoto i Yun'a	5
3.1.1 Specyfikacja podejścia	5
3.1.2 Funkcje wyjścia	6
3.1.3 Wybór konfiguracji manipulatora	6
3.1.4 Algorytm sterowania	6
3.1.5 Badania symulacyjne	7
3.2 Algorytm z rozszerzonymi funkcjami wyjściowymi	7
3.2.1 Specyfikacja podejścia	7
3.2.2 Funkcje wyjścia	7
3.2.3 Algorytm sterowania	7
3.2.4 Badania symulacyjne	8
4 Model satelity typu free-floating	10
5 Odsprzęganie wejściowo-wyjściowe dla satelity typu free-floating	10

Rozdział 1

Wstęp

1.1 Cel pracy

1.2 Zawartość pracy

Rozdział 2

Model manipulatora mobilnego

Manipulator mobilny to platforma mobilna z zamontowanym na niej manipulatorem. Biorąc pod uwagę typ ograniczeń nałożonych kolejno na platformę i manipulator można wydzielić 4 typy manipulatorów mobilnych:

1. typ (h,h) - platforma i manipulator związane są ograniczeniami holonomicznymi,
2. typ (nh,h) - na platformę nałożone są ograniczenia nieholonomiczne, natomiast na manipulator ograniczenia holonomiczne,
3. typ (nh,nh) - ograniczenia nałożone na platformę i manipulator są nieholonomiczne. Taki rodzaj manipulatora mobilnego został przedstawiony w [5],
4. typ (h,nh) - na manipulator nałożone są ograniczenia nieholonomiczne, natomiast na platformę holonomiczne.

W tym rozdziale zostanie przedstawiony model manipulatora mobilnego typu (nh,h) . W szczególności w rozważanym manipulatorze platformą będzie monocykl, czyli robot mobilny klasy (2,0). Za manipulator natomiast posłuży manipulator RTR. Rozważany manipulator mobilny przedstawiono na rys 1.

Przyjmijmy następujący wektor współrzędnych uogólnionych manipulatora mobilnego:

$$q = \begin{pmatrix} q_m \\ q_r \end{pmatrix} \in R^{n+p}, \quad (2.1)$$

gdzie $q_m = (x, y, \theta, \phi_1, \phi_2)^T \in R^n$ - wektor współrzędnych uogólnionych platformy mobilnej, $q_r = (q_1, q_2, q_3)^T \in R^p$ - wektor współrzędnych przegubowych manipulatora. Rozmiary przestrzeni wyniosą więc $n = 5$, $p = 3$.

2.1 Model we współrzędnych uogólnionych

Do wyprowadzenia równań manipulatora mobilnego wykorzystamy mechanikę Langrange'a. Lagranżjan L jest zdefiniowany jako różnica pomiędzy energią kinetyczną, a potencjalną układu

$$L(q, \dot{q}) = T(q, \dot{q}) - V(q). \quad (2.2)$$

Równania ruchu przy braku sił działających na system są następujące

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} = 0. \quad (2.3)$$

Lewą stronę równania (2.3) można przedstawić jako

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} = Q(q)\ddot{q} + C(q, \dot{q})\dot{q} + D(q). \quad (2.4)$$

Przy obecności sił zewnętrznych równanie (2.3) wygląda następująco

$$Q(q)\ddot{q} + C(q, \dot{q})\dot{q} + D(q) = F_e. \quad (2.5)$$

W naszym przypadku siły zewnętrze F_e są sumą 2 składowych: $F_e = B(q)u + F_{nh}$, gdzie $B(q)u$ to uogólnione siły wejściowe(tj. wywierane na system przez elementy wykonawcze), a F_{nh} to siły więzów nieholonomicznych(tj. siły zapewniające spełnienie ograniczeń nieholonomicznych).

Wektor $u = \begin{pmatrix} u_m \\ u_r \end{pmatrix} \in R^{(n+p-l)}$ jest wektorem sterowań.

Macierz $B(q) = \begin{bmatrix} B(q_m) & 0 \\ 0 & I_p \end{bmatrix} \in R^{(n+p) \times (n+p-l)}$ nazywana jest macierzą wejściową.

l jest liczbą ograniczeń nieholonomicznych.

Łatwo zauważyc, że jeśli obecne jest chociaż jedno ograniczenie nieholonomiczne macierz $B(q)$ nie jest macierzą kwadratową, przez co nie można jej odwrócić.

2.1.1 Ograniczenia nieholonomiczne

W naszym przypadku platformą manipulatora mobilnego jest monocykl. W założeniu ma poruszać się on bez poślizgu wzdluznego i poprzecznego kół. Aby zapewnić spełnienie tych założeń należy nałożyć trzy ograniczenia. Można pokazać, że jedynie dwa z nich to ograniczenia nieholonomiczne, a ostatnie można zcałkować do ograniczenia holonomicznego i w rezultacie wyeliminować jedną współrzędną z q_m [7]. Aby jednak pokazać możliwość algorytmu w działaniu przy w obecności ograniczeń holonomicznych i nieholonomicznych potraktujmy wszystkie trzy jednakowo i nie całkujmy żadnego z nich.

Przedstawmy ograniczenia w postaci Pfaffa

$$A(q)\dot{q} = 0, \quad (2.6)$$

Jako, że ograniczenia te dotyczą tylko platformy zapiszmy $A(q)$ następująco

$$A(q) = \begin{bmatrix} A(q_m) & 0 \end{bmatrix} \in R^{l \times (n+p)}. \quad (2.7)$$

Z [6] wiemy, że siła więzów nieholonomicznych równa jest $F_{nh} = A(q)^T \lambda$, gdzie λ to wektor mnożników Lagrange'a. Równanie (2.5) można więc teraz zapisać jako

$$Q(q)\ddot{q} + C(q, \dot{q})\dot{q} + D(q) = B(q)u + A(q)^T \lambda. \quad (2.8)$$

2.2 Model w prędkościach pomocniczych

Model postaci (2.8) posiada pewne wady: jak wspomniano w (2.1) macierz $B(q)$ nie daje się odwrócić z powodu obecności ograniczeń nieholonomicznych. W jawnej postaci występują również mnożniki Lagrange'a odpowiadające siłom tarcia statycznego zależnego od u, q, \dot{q}, t . Aby pozbyć się tych wad można przekształcić (2.8) do modelu wyrażonego w prędkościach pomocniczych.

Zauważmy najpierw, że ograniczenia (2.6) wymagają aby prędkość \dot{q} należała do jądra macierzy $A(q)$.

Niech baza jądra $A(q_m)$ będzie zbiorem wektorów $\{g_1(q_m), g_2(q_m), \dots, g_{n-l}(q_m)\}$. Zapiszmy prędkość \dot{q}_m w następujący sposób

$$\dot{q}_m = \sum_{k=1}^{n-l} g_k(q_m) \eta_k = \begin{bmatrix} g_1(q_m) & g_2(q_m) & \dots & g_{n-l}(q_m) \end{bmatrix} \begin{pmatrix} \eta_1 \\ \vdots \\ \eta_{n-l} \end{pmatrix} = G_m(q_m) \eta. \quad (2.9)$$

Wektor η nazywamy wektorem prędkości pomocniczych.

Zauważmy, że prędkość

$$\dot{q} = G(q)z, \quad (2.10)$$

gdzie $G(q) = \begin{bmatrix} G_m(q_m) & 0 \\ 0 & I_p \end{bmatrix} \in R^{(n+p) \times (n+p-l)}$ i $z = \begin{pmatrix} \eta \\ \dot{q}_r \end{pmatrix} \in R^{(n+p-l)}$ należy do jądra macierzy $A(q)$ i tym samym spełnia ograniczenia (2.6).

Możemy więc zapisać, że

$$A(q)G(q) = 0. \quad (2.11)$$

Biorąc transpozycję (2.11) otrzymamy

$$G^T(q)A^T(q) = 0. \quad (2.12)$$

Aby wyeliminować mnożniki Lagrange'a z (2.8) należy pomnożyć (2.8) lewostronnie przez G^T (dla czytelności pomińmy argumenty) i skorzystać z (2.12). Otrzymamy wtedy

$$G^T Q \ddot{q} + G^T C \dot{q} + G^T D = G^T B u + G^T A^T \lambda = G^T B u. \quad (2.13)$$

Zróżniczkujmy jeszcze (2.10) po czasie

$$\ddot{q} = \dot{G}z + G\dot{z}. \quad (2.14)$$

Po podstawieniu (2.10) i (2.14) do (2.13), otrzymamy model w prędkościach pomocniczych:

$$G^T Q(\dot{G}z + G\dot{z}) + G^T C G z + G^T D = G^T Q G \dot{z} + G^T (Q\dot{G} + CG)z + G^T D = G^T B u, \quad (2.15)$$

czyli

$$Q^* \dot{z} + C^* z + D^* = B^* u, \quad (2.16)$$

gdzie:

$$\begin{aligned} Q^* &= G^T Q G, \\ C^* &= G^T (Q \dot{G} + C G), \\ D^* &= G^T D, \\ B^* &= G^T B. \end{aligned}$$

Rozdział 3

Odsprzęganie wejściowo-wyjściowe dla manipulatora mobilnego

Algorytm odsprzęgania wejściowo-wyjściowego jest wykorzystywany do sterowania manipulatorów mobilnych. Rozwiązuje on zadanie śledzenia zadanej trajektorii efektora $y_d(t)$ w przestrzeni zewnętrznej. Algorytm odsprzęgania wejściowo-wyjściowego wymaga pełnej znajomości modelu manipulatora mobilnego.

Istnieją dwie wersje tego algorytmu:

- podstawowy algorytm Yamamoto i Yun'a [2][3]
- algorytm z rozszerzonymi funkcjami wyjściowymi[1]

Różnią się one wyborem funkcji wyjściowych $y(\cdot)$. Yamamoto i Yun założyli, że funkcje te będą zależne tylko od konfiguracji platformy q_m , natomiast Mazur pozwala aby funkcje wyjścia zależne były również od konfiguracji samego manipulatora q_r . Rozdział ten poświęcony jest opisowi obu podejść.

3.1 Podstawowy algorytm Yamamoto i Yun'a

3.1.1 Specyfikacja podejścia

Yamamoto i Yun założyli, że manipulator ma ustawić się w stałej konfiguracji o maksymalnej manipulowalności, a zadanie jest realizowane jedynie przez ruch platformy. Podejście takie traktuje cały manipulator mobilny bardziej jako samą platformę do której przymocowany jest unieruchomiony manipulator, niż pełen system, gdzie platforma i manipulator działają równocześnie.

3.1.2 Funkcje wyjścia

Przypomnijmy, że algorytm podstawowy wymaga aby funkcje wyjścia były zależne jedynie od konfiguracji platformy tj. $y(q_m)$. Wybór tych funkcji należy podporządkować rodzajowi zadania od zrealizowania. Do śledzenia trajektorii dwuwymiarowej (x_r, y_r)

3.1.3 Wybór konfiguracji manipulatora

Ponieważ manipulator pozostaje przez czas działania algorytmu nieruchomy i nie uczestniczy czynnie w realizacji zadania należy ustawić go w takiej konfiguracji, aby ułatwić platformie realizację zadania. Za przykładem Yamamoto i Yun'a wybierzmy konfigurację maksymalizującą manipulowalność. Manipulowalność manipulatora zdefiniowana jest jako [7]

$$w = \sqrt{\det(J(q_r)J^T(q_r))}, \quad (3.1)$$

gdzie $J(q_r)$ to jakobian manipulatora. Jeśli manipulator jest redundantny manipulowalność (3.1) można zredukować do

$$w = |\det(J(q_r))|. \quad (3.2)$$

3.1.4 Algorytm sterowania

Odpreszeganie i linearyzacja realizowane jest dzięki zastosowaniu do modelu (2.16) dwóch pętli sprzężenia zwrotnego. Pierwsza pętla - pętla wewnętrzna - przekształca układ do postaci liniowej typu "podwójny integrator", zaś druga - zewnętrzna - zapewnia realizację zadania tj. śledzenia trajektorii w uzyskanym układzie liniowym.

Prawo sterowania pierwszej pętli to prawo z rodziny algorytmów 'obliczanych momentów'

$$u = B^{*-1}[C^*z + D^* + Q^*v]. \quad (3.3)$$

Prawo drugiej pętli to

$$v = \Phi^{-1}[-\dot{\Phi}z + \zeta]. \quad (3.4)$$

Macierz Φ jest równa

$$\Phi(q_m) = \dot{y}(q_m) = \frac{dy}{dq_m}G_m.$$

Sprawdźmy teraz poprawność algorytmu. Zastosujmy najpierw pierwszą pętlę (3.3) do modelu (2.16). W wyniku otrzymamy

$$\dot{z} = v, \quad (3.5)$$

gdzie v jest nowym wejściem. Teraz zróżniczkujmy $y(q_m)$ dwukrotnie po czasie

$$\dot{y}(q_m) = \frac{dy}{dq_m}\dot{q}_m = \frac{dy}{dq_m}G_mz = \Phi(q)z, \quad (3.6)$$

$$\ddot{y}(q_m) = \dot{\Phi}z + \Phi\dot{z} = \dot{\Phi}z + \Phi v. \quad (3.7)$$

Łatwo sprawdzić, że po zastosowaniu prawa (3.4) do (3.7) otrzymamy

$$\ddot{y} = \zeta. \quad (3.8)$$

Wynikowy system (3.8) łatwo wysterować, na przykład za pomocą regulatora PD z ko-rektą

$$\zeta = \ddot{y}_d(t) - K_d \dot{e}(t) - K_p e(t), \quad (3.9)$$

gdzie $\ddot{y}_d(t)$ - druga pochodna zadanej funkcji wejściowej,
 $e(t) = y_d(t) - y(t)$, $\dot{e}(t) = \dot{y}_d(t) - \dot{y}(t)$ - błąd i jego pochodna,
 $K_d > 0$, $K_p > 0$ - nastawy regulacji.

Warunki działania algorytmu Ponieważ algorytm wykorzystuje odwrotność macie-rzy Φ należy zapewnić, aby

$$\det(\Phi) \neq 0 \quad (3.10)$$

3.1.5 Badania symulacyjne

3.2 Algorytm z rozszerzonymi funkcjami wyjściowymi

3.2.1 Specyfikacja podejścia

Odmienne od założeń Yamamoto i Yun'a rozszerzony algorytm nie wymaga bezruchu manipulatora. Dzięki temu platforma i manipulator mogą działać równolegle w sposób skoordynowany. Sterowanie platformy nie odbywa się w sposób bezpośredni, lecz działa w reakcji na dryf systemu.

3.2.2 Funkcje wyjścia

Funkcje wyjściowe mogą zależeć teraz od konfiguracji całego robota tj. i platformy i ma-nipulatora. Przyjmują one postacie[1]:

$$y(q) = \begin{pmatrix} y_1(q_m, q_r) \\ y_2(q_r) \end{pmatrix}, \quad (3.11)$$

gdzie $y_1(q_m, q_r)$ jest wektorem wybranych współrzędnych efektora w podstawowym ukła-dzie odniesienia (X_0, Y_0) , a $y_2(q_r)$ to wektor wybranych współrzędnych efektora w lokalnym układzie odniesienia (X_P, Y_P) związanym ze środkiem masy platformy.

Liczba współrzędnych efektora w układzie (X_0, Y_0) , które można odpsprząć zależy od liczby sterowań platformą. W przypadku platformy klasy $(2, 0)$ możemy śledzić dwie współrzędne.

3.2.3 Algorytm sterowania

Identycznie jak w przypadku algorytmu podstawowego, algorytm z rozszerzonymi funk-cjami wyjściowymi złożony jest z dwóch pętli sprzężeń zwrotnych[1]

- wewnętrznej pętli linearyzującej transformację wejście-stan
- zewnętrznej pętli linearyzującej transformację wejście-wyjście

Tak jak poprzednio wewnętrzna pętla wykorzystuje prawo sterowania z rodziny algoryt-mów 'obliczanych momentów'

$$u = B^{*-1}[C^*z + D^* + Q^*v]. \quad (3.12)$$

Zewnętrzna pętla ma także postać identyczną jak w przypadku algorytmu podstawowego

$$v = \Phi^{-1}[-\dot{\Phi}z + \zeta]. \quad (3.13)$$

Macierz Φ różni się jednak od poprzedniczki. Zależy ona od całej konfiguracji q , a nie tylko q_m

$$\Phi(q) = \begin{bmatrix} \Phi_m & \Phi_{mr} \\ 0 & \Phi_r \end{bmatrix} = \begin{bmatrix} \frac{\partial y_1}{\partial q_m} G_m & \frac{\partial y_1}{\partial q_r} \\ 0 & \frac{\partial y_2}{\partial q_r} \end{bmatrix}.$$

Prawo sterowania zewnętrznej pętli (3.13) wyprowadza się tak jak poprzednio. Oblicza się pochodną funkcji wyjściowej $y(q)$ po czasie

$$\dot{y}(q) = \begin{pmatrix} \frac{\partial y_1}{\partial q_m} \dot{q}_m + \frac{\partial y_1}{\partial q_r} \dot{q}_r \\ \frac{\partial y_2}{\partial q_r} \dot{q}_r \end{pmatrix} = \begin{bmatrix} \frac{\partial y_1}{\partial q_m} G_m & \frac{\partial y_1}{\partial q_r} \\ 0 & \frac{\partial y_2}{\partial q_r} \end{bmatrix} \begin{pmatrix} \eta \\ \dot{q}_r \end{pmatrix} = \Phi(q)z, \quad (3.14)$$

$\dot{y}(q)$ ponownie różniczkuję się po czasie:

$$\ddot{y} = \dot{\Phi}z + \Phi\dot{z}. \quad (3.15)$$

Stosując pierwszą pętlę (3.12) do modelu (2.16) mamy:

$$\dot{z} = v. \quad (3.16)$$

Po podstawieniu zależności (3.16) do równania (3.15) uzyskamy:

$$\ddot{y} = \dot{\Phi}z + \Phi v. \quad (3.17)$$

Aby zapewnić linearyzację wejściowo-wyjściową należy więc użyć wejścia:

$$v = \Phi^{-1}[-\dot{\Phi}z + \zeta]. \quad (3.18)$$

Po zastosowaniu algorytmu otrzymuje się więc funkcje wyjściowe w liniowej postaci:

$$\ddot{y} = \zeta. \quad (3.19)$$

Otrzymany system jest identyczny jak ten otrzymany po zastosowaniu algorytmu Yamamoto i Yun'a i również można nim sterować używając regulatora PD z korektą(3.9).

Warunki działania algorytmu Tak jak w przypadku podstawowego algorytmu Yamamoto i Yun'a należy zapewnić odwracalność macierzy Φ . Warunek jaki trzeba spełnić aby algorytm funkcjonował jest więc następujący

$$\det(\Phi) = \det(\Phi_m)\det(\Phi_r) \neq 0 \quad (3.20)$$

3.2.4 Badania symulacyjne

Rozważany monocykl posiada dwa wejścia (sterowania obu kół), można więc odsprząc dwie współrzędne efektora. Podczas badań postanowiono odsprząc współrzędne x i y efektora w podstawowym układzie odniesienia (X_0, Y_0). Funkcja wyjściowa $y_1(q_m, q_r)$ ma więc postać

$$y_1(q_m, q_r) = \begin{pmatrix} x + ac_0 + l_2 c_{01} + l_3 c_{01} c_3 \\ y + as_0 + l_2 s_{01} + l_3 s_{01} c_3 \end{pmatrix}.$$

Funkcja $y_2(q_r)$ to wektor współrzędnych x , y i z efektora w układzie środka masy platformy (X_P, Y_P)

$$y_2(q_r) = \begin{pmatrix} a + l_2 c_1 + l_3 c_1 c_3 \\ l_2 s_1 + l_3 s_1 c_3 \\ s_3 l_3 + q_2 \end{pmatrix}.$$

Początkowe położenie platformy było równe $q_m(0) = (x(0), y(0), \theta(0), \phi_1(0), \phi_2(0)) = (0, 0, 0, 0, 0)$, a początkowa konfiguracja manipulatora była równa $q_r(0) = (q_1(0), q_2(0), q_3(0)) = (\frac{\pi}{2}, 0.2, -\frac{\pi}{2})$. Prędkości początkowe $\dot{q}(0)$ były równe zeru.

Parametry geometryczne ustalone jako: $l_2 = 0.3\text{m}$, $l_3 = 0.2\text{m}$, $a = 0.2\text{m}$.

Funkcje wyjściowe miały więc następujące wartości początkowe

$$y(q_m(0), q_r(0)) = \begin{pmatrix} 0.2 \\ 0.3 \\ 0.2 \\ 0.3 \\ 0 \end{pmatrix}.$$

Wektor pochodnych \dot{y} równy był wektorowi zer.

Wybrano następujące trajektorie zadane $y_d(t)$

$$y_d(t) = \begin{pmatrix} 2t \\ -0.3 \cos(2t) \\ 0.5 \\ -0.3 \\ 0.2 \end{pmatrix}.$$

Błąd początkowy $e(t) = y_d(t) - y(t)$ i jego pochodna były więc równe

$$e(0) = \begin{pmatrix} -0.2 \\ -0.6 \\ 0.3 \\ -0.6 \\ 0.2 \end{pmatrix}, \quad \dot{e}(0) = \begin{pmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

W badaniach wykorzystano środowisko Matlab/Simulink.

Rozdział 4

Model satelity typu free-floating

Rozdział 5

Odsprzeganie wejściowo-wyjściowe dla satelity typu free-floating

Bibliografia

- [1] Mazur A. *New approach to designing input-output decoupling controllers for mobile manipulators*. Bull. of the Polish Academy of sciences Tech. Sci. 53(1):31-37, (2005)
- [2] Yamamoto Y, Yun X. *Coordinating locomotion and manipulation of a mobile manipulator*. IEEE Trans. on Automatic Control, 39(6):1326-1332, (1994)
- [3] Yamamoto Y, Yun X. *Effect of the dynamic interaction on coordinated control of mobile manipulators*. IEEE Trans. Robotics Automat, 12(5):816-824, (1996)
- [4] Mazur A, Arent K. *Lecture Notes in Control and Information Sciences*, 335:55-71 (2007)
- [5] Tchoń K, Jakubiak J. *Acceleration-Driven Kinematics of Mobile Manipulators: An Endogenous Configuration Space Approach*, pp.(469-476), (2004)
- [6] M. R. Flannery. *The Enigma of Nonholonomic Constraints*. American Association of Physics Teachers, 73(3):265–272, (2005).

- [7] Yamamoto Y, Yun X. *Coordinating locomotion and manipulation of a mobile manipulator*. In Proceedings of 31st IEEE Conference on Decision and Control, pp.(2643-2648), (1992)