DOI 10.15199/148.2018.5.2

Metoda samoorganizacji i podążania za liderem roju nieholonomicznych robotów mobilnych z wykorzystaniem wirtualnych elementów sprężysto-tłumiących

Self-organization and leader following method of nonholonomic mobile robots based on virtual springs and dampers

VICTOR EREMEYEV
JAKUB WIECH

Celem algorytmu sterowania rojem jest osiągnięcie pożądanego zachowania roju przez lokalną interakcję między robotami. Zachowanie roju jest efektem złożenia wszystkich decyzji podejmowanych przez roboty zgodnie z prostymi zasadami. Przy podejmowaniu decyzji roboty określają swoją pozycję względem swoich sąsiadów i otaczającego ich środowiska, które mogą wykryć za pośrednictwem sensorów. Kolejnym wymaganiem jest skalowalność roju, która musi być zapewniona na etapie projektowania algorytmu sterowania rojem. Dostępne metody sterowania rojem, które spełniają wszystkie wymagania, są często inspirowane biologią, oparte na zjawiskach fizycznych lub różnych algorytmach optymalizacyjnych. Istnieją podobne prace oparte na wirtualnych sprężynach. Na przykład Urcola i in. [1] przedstawia kontrolę formacji opartą na wirtualnej strukturze składającej się z wirtualnych tłumików i sprężyn łączących roboty podążające za liderem. Shucker i Bennett [2] proponują całkowicie rozproszony robotyczny makrosensor, którego algorytm sterowania był wzorowany połączeniami w sprężystej siatce. Innym przykładem jest rój satelitów łączonych wirtualnymi sprężynami i tłumikami zaproponowany przez Chen i in. [3], w którym zbadano formację satelitów w polu grawitacyjnym. Ponadto Belkacem i Foudil [4] przeprowadzili badania modelu agregacji roju, opierając się na równowadze robotów wynikającej z połączenia wirtualnymi sprężynami i tłumikami.

W prezentowanym rozwiązaniu Autorzy proponują własną metodę wyznaczania parametrów ruchu robota na podstawie działających wirtualnych sił. Do sterowania ruchem robota nieholonomicznego zdefiniowano zależność między wypadkową wirtualną siłą działającą na robota a jego prędkością liniową i prędkością kątową obrotu ramy robota.

STRESZCZENIE

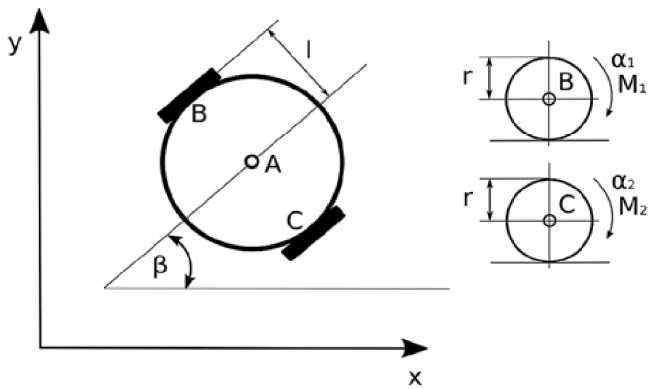
Celem pracy jest demonstracja metody samoorganizacji i podążania za liderem nieholonomicznego roju robotów mobilnych, opartej na wirtualnych, tłumionych, liniowych sprężynach łączących sąsiadujące roboty. Analizę metody sterowania poprzedza wyprowadzenie dynamiki dwukołowego robota oraz określenie zależności między wirtualnymi siłami a wejściami sterującymi robota w celu osiągnięcia stabilnej formacji roju. Analizowane są dwa przypadki sterowania rojem. W pierwszym przypadku spójność roju jest osiągnięta przez wirtualne sprężyny z tłumikami, łączące najbliższe roboty bez wyznaczonego lidera. W drugim przypadku wprowadzany jest lider roju oddziałujący wirtualnymi siłami na najbliższych i drugich sąsiadów, umożliwiając podążanie roju za liderem. Praca kończy się symulacjami numerycznymi oceniającymi wydajność zaproponowanej metody sterowania rojem.

Słowa kluczowe: rój, roboty mobilne, metoda wirtualnych sprężyn i tłumików, sterowanie formacją, podążanie za liderem, samoorganizacja.

ABSTRACT

This paper presents a method for self-organization and leader following of nonholonomic robotic swarm based on virtual spring damper mesh. The dynamics of two wheel robot is derived using Euler-Lagrange's method and relation between virtual forces and robot control inputs is defined in order to establish stable desired swarm formation. We analyze two cases of swarm control. In the first case the swarm cohesion is achieved without designated leader by virtual spring damper mesh connecting nearest neighboring robots. In the second case we introduce a swarm leader interacting with nearest and second neighbors allowing the swarm to follow the leader. Numeric simulation results are presented to illustrate the performance of the proposed control method.

Keywords: swarm, mobile robots, virtual spring damper method, formation control, lider following, self-organization



Rys. 1. Schemat robota dwukołowego

Dynamika robota dwukołowego

Najczęściej używanymi robotami w badaniach rojów robotów są nieholonomiczne roboty dwukołowe. Schemat robota jest pokazany na rys. 1.

Aby opisać dynamikę robota dwukołowego, można zastosować równania *Eulera-Lagrange'a* z mnożnikami. Niestety, jedną z wad jest konieczność oddzielenia mnożników od momentów obrotowych. Aby uniknąć oddzielania mnożników, użyto innego formalizmu przy zastosowaniu równań *Maggiego* [5], określonych równaniem:

$$\sum_{j=1}^{n} C_{ij}\left[\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial E}{\partial \dot{q}_i}\right)-\frac{\partial E}{\partial q_i}\right]=\Theta_i,\ i=1,\ldots s,\ j=1,\ldots,n \tag{1}$$

gdzie: s jest liczbą niezależnych parametrów w uogólnionych współrzędnych q_j równą liczbie stopni swobody. Prędkości uogólnione i prawe strony równań (1) są opisane przez relacje:

$$\dot{q}_i=\sum_{i=1}^{s} C_{ij}\dot{e}_i+G_j \tag{2}$$

$$\sum_{i=1}^{s}\Theta_i\delta e_i=\sum_{i=1}^{s}\delta e_i\sum_{j=1}^{n}C_{ij}Q_j \tag{3}$$

gdzie: $\dot{e}_i$ nazywane są parametrami kinetycznymi i w tym przypadku są równe $\dot{\alpha}_1,\dot{\alpha}_2$.

Współrzędne uogólnione i prędkości uogólnione dla robota dwukołowego wynoszą:

$$\begin{aligned} q&=[x_A,y_A,\beta,\alpha_1,\alpha_2]^T\\ \dot{q}&=[\dot{x}_A,\dot{y}_A,\dot{\beta},\dot{\alpha}_1,\dot{\alpha}_2]^T \end{aligned} \tag{4}$$

Opisany robot ma dwa stopnie swobody, niezależne współrzędne uogólnione są kątami obrotu kół α_1,α_2. Za pomocą analizy kinematyki robota można określić prędkości w następujący sposób:

$$\begin{aligned} &\dot{\beta}=\tfrac{r}{2l}(\dot{\alpha}_1-\dot{\alpha}_2),\ \dot{x}_A=\tfrac{r}{2}(\dot{\alpha}_1+\dot{\alpha}_2)\cos\beta,\\ &\dot{y}_A=\tfrac{r}{2}(\dot{\alpha}_1+\dot{\alpha}_2)\sin\beta \end{aligned} \tag{5}$$

Całkowita energia kinetyczna E jest równa sumie energii kinetycznej obu kół i ramy robota.

$$E=Ew_1+Ew_2+E_r \tag{6}$$

Energia kinetyczna kół i ramy robota wyrażona jest przez równania:

$$\begin{aligned} E_r&=\frac{1}{2}m_r v_A{}^2+\frac{1}{2}I_{zr}\beta^2\\ E_{w1}&=\frac{1}{2}m_w v_B{}^2+\frac{1}{2}I_{zw}\dot{\alpha}_1{}^2+\frac{1}{2}I_{xw}\dot{\beta}^2\\ E_{w2}&=\frac{1}{2}m_w v_C{}^2+\frac{1}{2}I_{zw}\dot{\alpha}_2{}^2+\frac{1}{2}I_{xw}\dot{\beta}^2 \end{aligned} \tag{7}$$

gdzie: m_w, m_r, I_{zw}, I_{xw}, I_{zr} są masami i masowymi momentami bezwładności kół i ramy robota w osiach x i z, r jest promieniem koła, l jest odległością między punktem A a środkami koła B i C.

Używając równań (2) i (5), można obliczyć współczynniki C_{ij}, G_j, ponadto, dzięki równaniom (1) i (3), można określić siły uogólnione Q_j w następującej postaci:

$$\begin{aligned} &C_{11}=\tfrac{r}{2}\cos\beta,\ C_{21}=\tfrac{r}{2}\cos\beta,\ G_1=0,\ Q_1=(2m_w+m_r)\ddot{x}_A\\ &C_{12}=\tfrac{r}{2}\cos\beta,\ C_{22}=\tfrac{r}{2}\cos\beta,\ G_2=0,\ Q_2=(2m_w+m_r)x\ddot{y}_A\\ &C_{13}=\tfrac{r}{2l},\ C_{23}=-\tfrac{r}{2l},\ G_3=0,\ Q_3=2m_w l^2\ddot{\beta}+2I_{zw}\left(\tfrac{l}{r}\right)^2\ddot{\beta}+2I_{xw}\ddot{\beta}+I_{zr}\ddot{\beta}\\ &C_{14}=1,\ C_{24}=0,\ G_4=0,\ Q_4=I_{zw}\ddot{\alpha}_1\\ &C_{15}=0,\ C_{25}=1,\ G_5=0,\ Q_5=I_{zw}\ddot{\alpha}_2 \end{aligned} \tag{8}$$

Siły uogólnione i współczynniki C_{ij}, G_j w (8) oraz równaniu (1) prowadzą do następujących równań *Maggiego*:

$$\begin{aligned} &(2m_w+m_r)\left(\frac{r}{2}\right)^2(\ddot{\alpha}_1+\ddot{\alpha}_2)+(2m_w l^2+2I_{xw}+I_{zr})\\ &\quad\tfrac{r}{2l}(\ddot{\alpha}_1-\ddot{\alpha}_2)+I_{zw}\ddot{\alpha}_1=M_1-N_1\\ &(2m_w+m_r)\left(\frac{r}{2}\right)^2(\ddot{\alpha}_1+\ddot{\alpha}_2)-(2m_w l^2+2I_{xw}+I_{zr})\\ &\quad\tfrac{r}{2l}(\ddot{\alpha}_1-\ddot{\alpha}_2)+I_{zw}\ddot{\alpha}_2=M_2-N_2 \end{aligned} \tag{9}$$

Jeżeli zapiszemy równanie (9) w postaci macierzowej, otrzymamy:

$$M\ddot{q}+F(\dot{q})=U \tag{10}$$

gdzie: M to macierz bezwładności, $F(\dot{q})$ to wektor oporów ruchu, U jest wektorem momentów napędowych kół robota.

$$\begin{aligned} &M=\begin{bmatrix} a_1+a_2+a_3 & a_1-a_2\\ a_1-a_2 & a_1+a_2+a_3\end{bmatrix},\ F(\dot{q})=\begin{bmatrix} a_5\,\mathrm{sgn}\,\dot{\alpha}_1\\ a_6\,\mathrm{sgn}\,\dot{\alpha}_2\end{bmatrix},\\ &U=\begin{bmatrix}M_1 & M_2\end{bmatrix}^T \end{aligned} \tag{11}$$

Parametry a_1, ..., a_5 określamy według zależności:

$$\begin{aligned} &a_1=(2m_w+m_r)\left(\frac{r}{2}\right)^2,\ a_2=(2m_w l^2+2I_{xw}+I_{zr})\frac{r}{2l}\\ &a_3=I_{zw},\ a_4=N_1 f_1,\ a_5=N_2 f_2 \end{aligned} \tag{12}$$

gdzie: N_1, N_2 to siły nacisku na koła 1 i 2, f_1, f_2 to współczynniki tarcia tocznego kół.

Wejścia sterujące robotem to przyśpieszenia kątowe kół $\ddot{q} = [\ddot{\alpha}_1, \ddot{\alpha}_2]^T$ oraz prędkości kątowe kół $\dot{q} = [\dot{\alpha}, \dot{\alpha}]^T$ określone równaniami:

$$\dot{\alpha}_1 = \frac{v_A}{r} + \dot{\beta}\frac{l}{r}, \; \dot{\alpha}_2 = \frac{v_A}{r} - \dot{\beta}\frac{l}{r} \tag{13}$$

gdzie: v_A to prędkość punktu A ramy robota, $\dot{\beta}$ to prędkość kątowa obrotu ramy robota.

Algorytm sterowania rojem

Roboty w roju połączone są wirtualnymi sprężynami i tłumikami połączonymi równolegle. Zakładamy, że roboty mają masę m i są połączone za pomocą sprężyn o sztywności k i tłumikach o współczynniku tłumienia c. Długość spoczynkowa sprężyn jest równa d_0.

Końce wirtualnych sprężyn i tłumików umieszczone są w środkach mas robotów, przykładowa konfiguracja połączeń wirtualnych widoczna jest na rys. 2. Środek masy oznaczony jest jako punkt A na rys. 1.

Każde wirtualne połączenie wywiera wirtualną siłę równą F_{ij} sumie wirtualnych sił pochodzących od sprężyny i tłumika. Zakładamy, że wirtualna sprężyna i tłumik są liniowe, a ich siły są proporcjonalne odpowiednio do deformacji sprężyny i względnej prędkości między końcami sprężyny.

$$F_{ij} = F_{Sij} + F_{Dij} = k(d_{ij} - d_0) + c\frac{\mathrm{d}}{\mathrm{d}t}(d_{ij} - d_0) \tag{14}$$

gdzie: k to współczynnik sprężystości sprężyny, c to współczynnik tłumienia, d_{ij} to aktualna długość sprężyny, d_0 to długość spoczynkowa sprężyny.

Każdy robot ma ograniczony zasięg sensorów SR_r, który określa, ile połączeń wirtualnych jest tworzonych, ograniczając liczbę połączeń do liczby widocznych sąsiadujących robotów. Aby utworzyć połączenia tylko z najbliższymi sąsiadami, zakres sensorów musi spełniać nierówność:

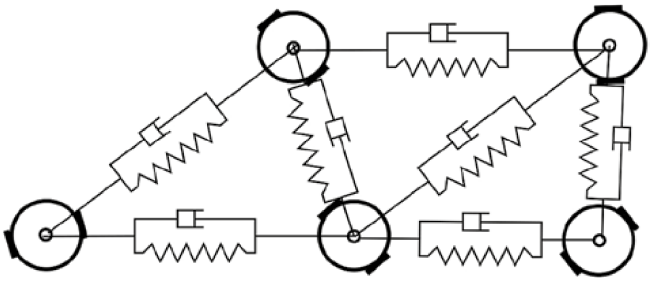
$$d_0\sqrt{3} \rangle SR_r \rangle d_0 \tag{15}$$

Na przykład ustawianie zakresu sensorów w wartości 1.2 d_0 powoduje utworzenie wirtualnych połączeń pomiędzy najbliższymi sąsiadami tworzącymi rój, zbudowanych z trójkątów równobocznych pokazanych na rys. 3.

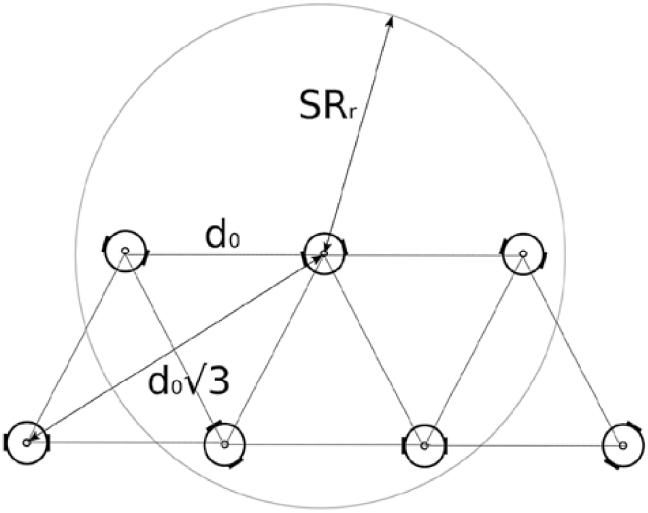
Całkowita wirtualna siła wywierana na robota i jest równa sumie wszystkich wirtualnych sił między robotem i a jego widocznymi sąsiadami N_r.

$$\overline{F}_i = \sum_{j \in N_r} \overline{F}_{ij} = \sum_{j \in N_r} \overline{F}_{Sij} + \overline{F}_{Dij} \tag{16}$$

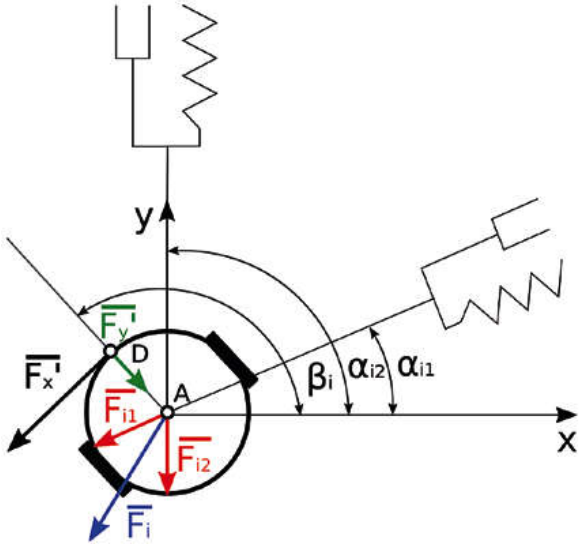
Z równania (13) można wywnioskować, że potrzebna jest wartość prędkości liniowej v_A oraz prędkość kątowa obrotu ramy $\dot{\beta}$, aby określić wartości wejściowe sterowania robota, znając wartość wirtualnej siły wypadkowej $\overline{F}_i$. Aby określić wartości v_A, $\dot{\beta}$ należy najpierw znaleźć odpowiednie składowe siły wypadkowej powodujące ruch robota do przodu lub do tyłu (składowa siła F_y') oraz obrót robota wokół punktu A (składowa siła F_x'). Wymienione siły pokazano na rys. 4.



Rys. 2. Sposób połączenia robotów wirtualnymi sprężynami I tłumikami



Rys. 3. Zakres sensorów w połączeniach z najbliższymi sąsiadami



Rys. 4. Wirtualne siły powodujące ruch robota i

Punktem zaczepienia wektorów $\bar{F}_y', \bar{F}_x'$ jest punkt D umieszczony na przodzie robota w odległości l od punktu A. Jeśli znamy wartości wirtualnych sił pochodzących ze wszystkich połączeń wirtualnych i ich orientację, możemy znaleźć składowe x i y całkowitej siły $\bar{F}_i$ wyrażone równaniami:

$$F_x = \sum_{j \in N_r} \bar{F}_{ij} \cos \alpha_{ij}, \qquad F_y = \sum_{j \in N_r} \bar{F}_{ij} \sin \alpha_{ij} \tag{17}$$

Wektory $\bar{F}_y', \bar{F}_x'$ są składowymi wektora F_{ij} w obracającym się układzie współrzędnych wokół punktu A o kąt $\varphi_i = \beta_i - \frac{\pi}{2}$. Dodanie kąta $\frac{\pi}{2}$ kieruje wektor $\bar{F}_y'$ zamiast wektora $\bar{F}_x'$ w kierunku wektora prędkości $\bar{v}_A$. Wartości wektorów $\bar{F}_y', \bar{F}_x'$, są wyliczane według zależności:

$$F_x' = -F_x \sin \varphi_i + F_y \cos \varphi_i, \qquad F_y' = F_x \cos \varphi_i + F_y \sin \varphi_i \tag{18}$$

Po zastosowaniu równań (18) wartości prędkości liniowej v_A oraz prędkości kątowej ramy $\dot{\beta}$ są określone przez równania:

$$v_A = \int \frac{F_y'}{2m_w + m_r} \mathrm{d}t, \qquad \dot{\beta} = -\lambda F_x' \tag{19}$$

gdzie λ jest stałą wzmocnienia.

Stała wzmocnienia λ umożliwia sterowanie prędkością obrotową ramy robota wokół punktu A, ustawiając ramę robota w kierunku wirtualnej siły wypadkowej $\bar{F}_i$ działającej na robota. Ponadto, współczynnik tłumienia c i współczynnik sztywności k mogą być ustalone przez projektanta tak, że wartości v_A i $\dot{\beta}$ są odpowiednie dla danego zadania roju.

Symulacja

Aby ocenić wydajność przedstawionej metody sterowania, przeprowadzono dwie symulacje. Roboty były początkowo losowo rozstawione z losowymi początkowymi kątami obrotu ramy. W obu symulacjach użyto 21 robotów, w pierwszym przypadku rój samoorganizował się bez lidera, tworząc formację z trójkątów równobocznych, wyniki pokazano na rys. 5. W drugim przypadku jeden z robotów został wyznaczony jako lider roju, tworzący wirtualne połączenia z najbliższymi i drugimi sąsiadami. Umożliwia to samoorganizację roju w postaci okręgów wokół lidera. Parametry i stałe algorytmu sterowania dla obu symulacji są wymienione w tabeli.

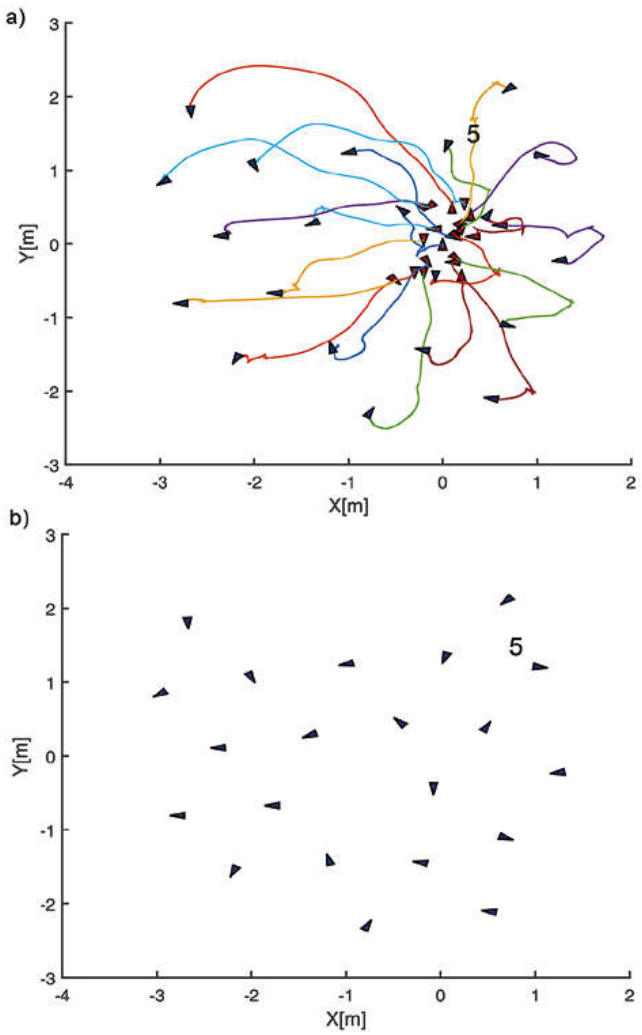
Po 7 s prędkości liniowe i kątowe robotów zaczęły oscylować wokół zera, co przy obecnym tarciu i ograniczonym czasie odpowiedzi silników robota osiągnie zero w skończonym czasie. W pierwszej symulacji formacja utworzona przez rój nie jest pozbawiona luk, co może sugerować zmianę parametrów symulacji na inne wartości.

Siły z wirtualnych sprężyn i tłumików działające na roboty również oscylują wokół zera lub w niektórych przypadkach osiągnęły zero. W pierwszej symulacji w przypadku robota nr 5, oznaczonego na rys. 5, prędkości oraz wirtualne siły działające na robota oscylują wokół zera, podczas gdy kąt obrotu ramy ma wartość niezerową (rys. 6).

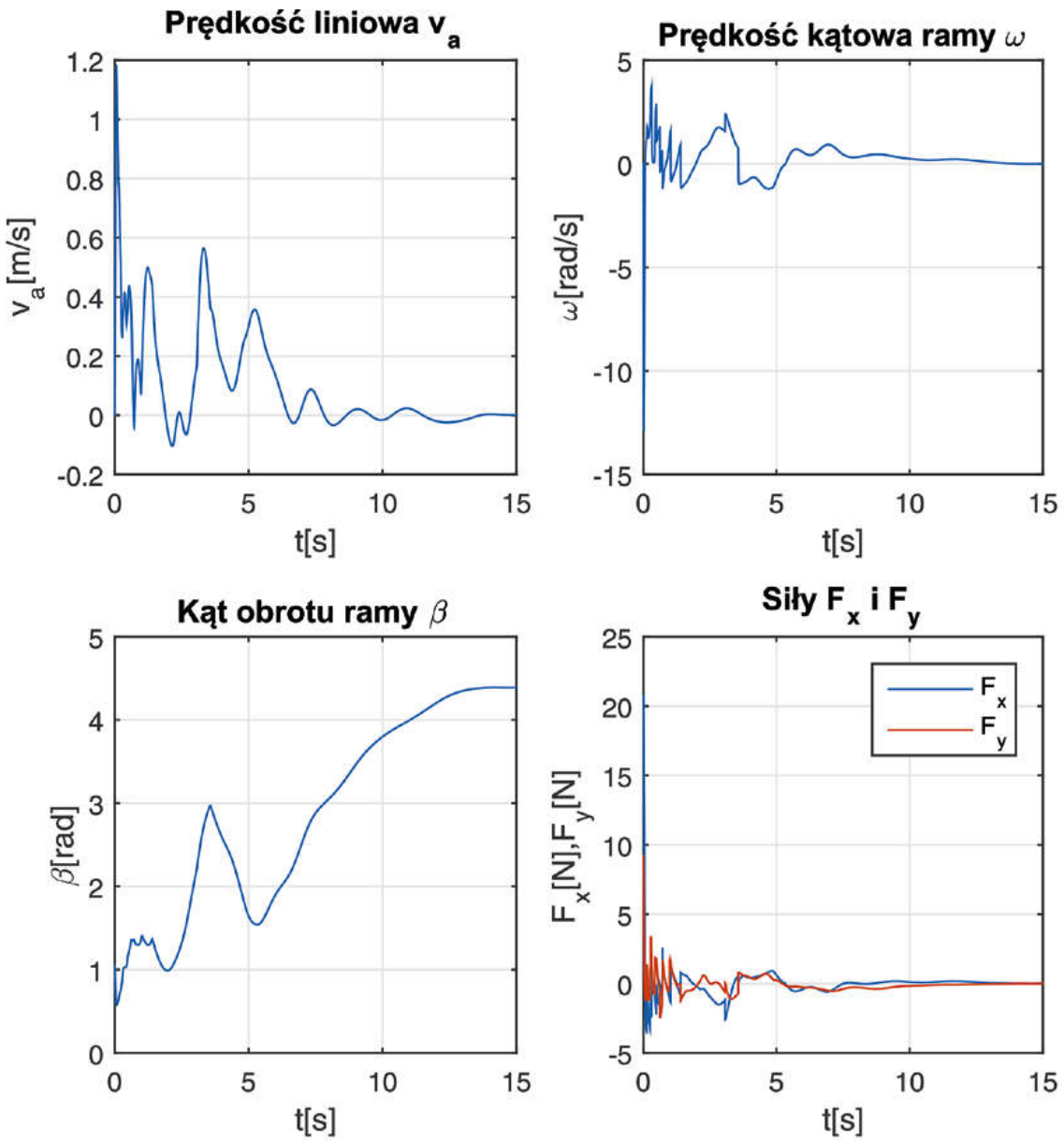
Oscylacje w pierwszych 2,5 s są obecne z powodu usunięcia wirtualnych połączeń z robotami wychodzącymi poza zasięg sensorów. Proces ten powoduje nagłą zmianę wartości wirtualnej siły wypadkowej, a także jej kierunku, co powoduje zmianę prędkości v_A i $\dot{\beta}$ robota.

Parametry symulacji

m_w, kg	m_r, kg	I_{zr}, kgm²	I_{xw}, kgm²	I_{zw}, kgm²	N, N	l, m	r, m	f, m	k, N/m	c, Ns/m	λ	d_0, m
0,05	0,4	0,08	0,01	0,03	22	0,058	0,032	0,004	7	1	4	1
0,05	0,4	0,08	0,01	0,03	22	0,058	0,032	0,004	15	1,5	4	1



Rys. 5. Samoorganizacja roju (bez lidera) dla pierwszego zestawu parametrów: a) trajektorie robotów, b) końcowe pozycje robotów



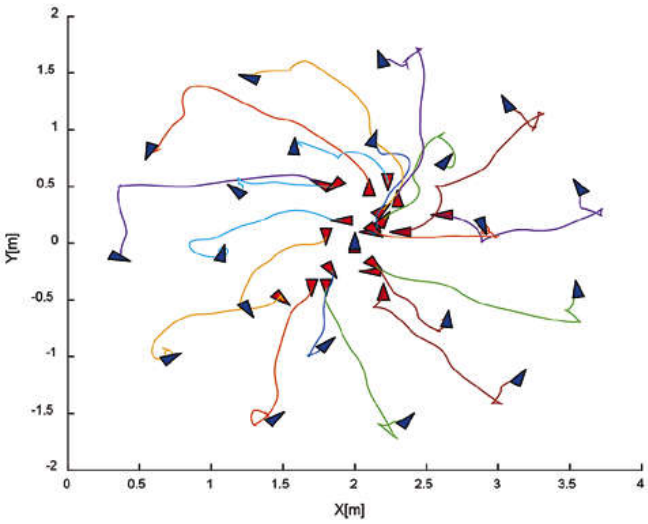
Rys. 6. Wyniki symulacji dla robota nr 5

Samoorganizacja roju z liderem

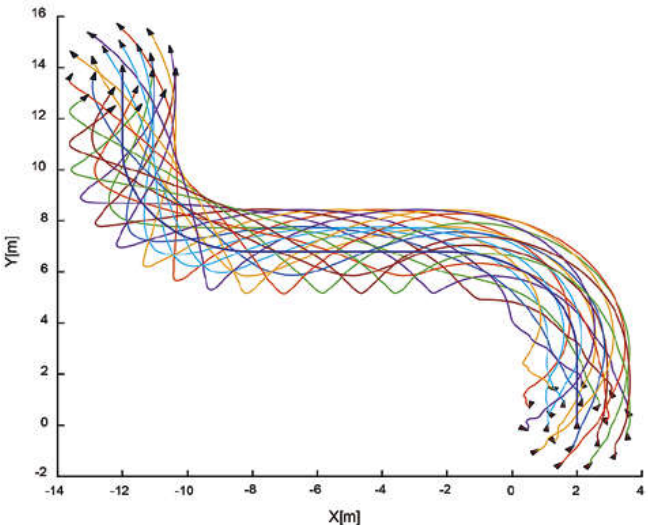
Spójność roju osiągana jest dzięki dwóm rodzajom interakcji. Interakcja między sąsiadującymi robotami, jak w poprzednim przykładzie, i dodatkowe siły wirtualne działające od lidera do najbliższych i drugich sąsiadów. Przy obliczaniu wirtualnych sił wybrano jednakowe wartości *k*, *c* (patrz tabela). Długość spoczynkowa sprężyny między liderem a najbliższymi sąsiadami jest równa d_0, a między liderem i drugimi sąsiadami wynosi 1,5 d_0. Średnica roju mieści się pomiędzy $2d_0$ i d_0. W tym przypadku jest równa 1,57 m.

Roboty podążające za liderem rozstawiają się wokół lidera, tworząc dwa okręgi widoczne na rys. 7.

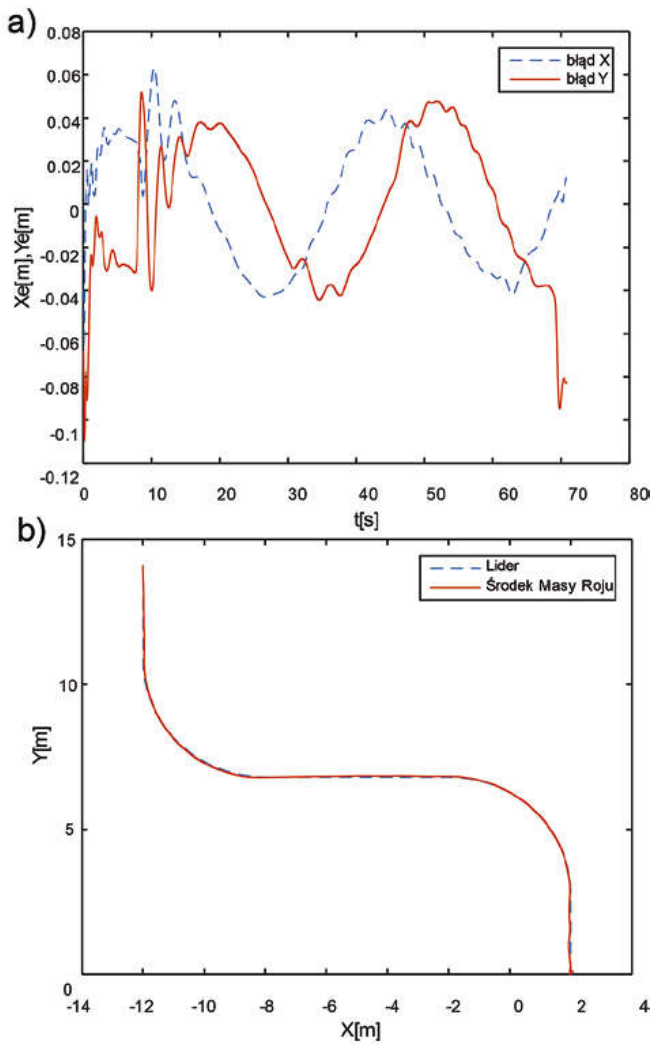
Wybrany lider podąża według założonej trajektorii, natomiast pozostałe roboty po tak osiągniętej formacji podążają za trajektorią lidera. Przykład podążania za liderem pokazano na rys. 8.



Rys. 7. Samoorganizacja roju z liderem



Rys. 8. Rój podążający za liderem



Rys. 9. Porównanie trajektorii środka masy roju z trajektorią lidera: a) błąd x i y, b) trajektorie lidera i środka masy roju

Podczas ruchu roju rój utrzymuje swoją spójność. Rój porusza się w tym samym kierunku co lider. Poszczególne roboty obracają się wokół lidera w lewo, czyli w tym samym kierunku, co pierwszy obrót roju. Podążające roboty kontynuują rotację ze stałą prędkością aż do następnego obrotu roju w prawo, co spowalnia rotację robotów. Trajektoria roju może być reprezentowana jako trajektoria środka masy roju i obliczana tylko dla podążających robotów oraz porównywana z trajektorią lidera.

Różnica między pozycją lidera roju a środkiem masy pokazana na rysunku 9a jest mniejsza niż 4 cm na kierunku *x* i *y*. Obliczony błąd może być użyty jako parametr projektowy do wyboru poprawnych wartości k, c, λ i d_0 dla podążania roju za liderem. Wykresy trajektorii lidera i środka masy roju przedstawiono na rys. 9 b.

Wnioski i przyszłe kierunki pracy

W artykule przedstawiono metodę samoorganizacji nieholonomicznego roju robotów za pomocą wirtualnych tłumionych sprężyn. Po wybraniu odpowiednich wartości sztywności sprężyny, współczynnika tłumienia i stałej wzmocnienia λ, rój jest w stanie uzyskać formację składającą się z trójkątów równobocznych. Co więcej, jeśli wprowadzimy lidera roju, przy prawidłowych parametrach projektowych, rój podąża za nim, utrzymując osiągniętą formację w procesie samoorganizacji. Symulacje pokazują, że dla roju nieholonomicznych robotów można zaprojektować algorytm samoorganizacji oparty na wirtualnych siłach sprężyn i tłumików. Przyszłe badania mogą dotyczyć problemu wyboru optymalnych wartości współczynników k, c, λ i d_0 w celu uzyskania lepszej wydajności algorytmu. Ponadto, przyszłe prace obejmują opracowanie rozszerzonej wersji prezentowanej metody z uwzględnieniem omijania przeszkód i kontrolą kształtu formacji roju.

Artykuł recenzowany

LITERATURA

[1] Urcola P., L. Riazuelo, M. Lazaro, L. Montano. 2008. "Cooperative navigation using environment compliant robot formations" [in:] *IEEE/RSJ International Conference on Intelligent Robots and Systems*: 2789 – 2794.
[2] Shucker B., J. K. Bennett. 2005. "Virtual spring mesh algorithms for control of distributed robotic macrosensors". University of Colorado at Boulder, Technical Report CU-CS-996-05.
[3] Chen Q., S. M. Veres, Y. Wang, Y. Meng. 2015. "Virtual spring, -damper mesh-based formation control for spacecraft swarms in potential fields". *J. Guid. Control Dyn.* 38(3): 539 546.
[4] Balkacem K., C. Foudil. 2016. "A virtual viscoelastic based aggregation model for self-organization of swarm robots system". [In] *TAROS 2016: Towards Autonomous Robotic Systems*: 202 – 213, Springer, Cham.
[5] Giergiel J., W. Żylski. 2005. "Description of motion of a mobile robot by Maggie's equations". *Journal of theoretical and applied mechanics* 43 (3): 511 521.

dr hab. Victor Eremeyev, prof. PG
Politechnika Gdańska,
Wydział Inżynierii Lądowej i Środowiska
e-mail: victor.eremeev@pg.edu.pl
mgr inż. Jakub Wiech
Katedra Mechaniki Stosowanej i Robotyki,
Politechnika Rzeszowska im. Ignacego Łukasiewicza
al. Powstańców Warszawy 8, 35-959 Rzeszów
e-mail: j.wiech @prz.edu.pl