

Andrzej BURGHARDT

POLITECHNIKA RZESZOWSKA, WYDZIAŁ BUDOWY MASZYN, KATEDRA MECHANIKI STOSOWANEJ I ROBOTYKI

Sterowanie behawioralne minirobota kołowego.

mgr inż. Andrzej BURGHARDT

Ukończył studia na Wydziale Inżynierii Mechanicznej i Robotyki AGH w 2001 roku o specjalności Robotyka i Mechatronika. Obecnie pracuje w Katedrze Mechaniki Stosowanej i Robotyki Politechniki Rzeszowskiej na stanowisku asystenta. Jego zainteresowania skupiają się na zagadnieniach prototypowania i implementacji układów sterowania mobilnych robotów kołowych.



Streszczenie

W artykule przedstawiono propozycję realizacji elementarnych zachowań, mobilnego minirobota, typu „idź do celu” i „znajdź środek wolnej przestrzeni”. W prezentowanym rozwiązaniu zastosowano hierarchiczny, układ sterowania. W ramach niskiego poziomu sterowania ruchem nadążnym mobilnego robota kołowego zastosowano adaptacyjny układ realizujący zadaną trajektorię. Generację trajektorii zdeterminowaną warunkami zewnętrznymi uzyskano z nadrzędnego neuronowego układu sterowania. Symulację zaproponowanych rozwiązań przeprowadzono w pakiecie, Matlab/Simulink. Ponadto zaproponowano ich połączenie z wykorzystaniem sieci neuronowych typu WTA, realizując w ten sposób zachowanie „idź do celu z ominięciem przeszkód”. Uzyskane rozwiązania zweryfikowano na obiekcie rzeczywistym w zaproponowanym przez autora środowisku szybkiego prototypowania.

Abstract

Expansion of the range of robot task and an increase in robot autonomy created a need to generate trajectories on-line to avoid collisions with obstacles in the workspace while approaching a given goal point. In this paper we discuss two main behaviors: "a reaching the middle of a collision-free space", "a goal-seeking". For this purposes we propose using hierarchical controllers which include low-level controllers and upper-level controllers. As low-level controller we use an adaptive controller which enables trajectory tracking. It is generated in accordance to one of the elementary behaviors considering unknown environment. For this purpose we use algorithm based on Braitenberg's principles modified for nonholonomic mobile robot. All simulations are made using emulator of mobile robot created in Matlab/Simulink. To verify working of mentioned algorithm on the real object there was created new testing station based on software Matlab 6.5/Simulink 5.0 applying additional toolbox Real Time Windows Target 2.2. LPT port enables communication between controller of mobile robot and control algorithm in RTWT. The proposed approach enables generating and tracking trajectory for mobile robot in real time. The results obtained in all simulations confirmed effectiveness of path planning and control strategy.

1. Wstęp

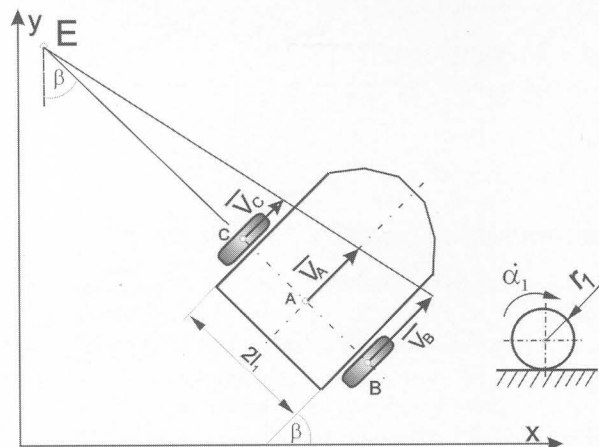
Rozwój zakresów zastosowań robotów mobilnych stworzył zapotrzebowanie na systemy sterowania generujące trajektorię w czasie rzeczywistym, np.: robot modyfikuje trajektorię on-line w celu ominięcia przeszkód, które znalazły się w jego przestrzeni roboczej podczas realizacji określonego zadania. W pracy zostanie przedstawiona propozycja rozwiązania problemu realizacji elementarnych zachowań typu „idź do celu”, „znajdź środek wolnej przestrzeni”, których poprawna realizacja oraz odpowiednie połączenie jest jednym ze sposobów rozwiązania wcześniej sformułowanego problemu. Tematyka ta będzie rozwiązana przez zastosowanie hierarchicznego, inteligentnego układu sterowania, rozumianego jako układu sterowania składającego się z niskiego i nadrzędnego poziomu sterowania.

W ramach niskiego poziomu sterowania ruchem nadążnym mobilnego robota kołowego rozważany będzie adaptacyjny układ sterowania realizujący zadaną trajektorię. Generację trajektorii zdeterminowaną warunkami zewnętrznymi uzyskano z nadrzędnego

neuronowego układu sterowania, którego synteza oparta została o ideę algorytmu Braitenberga [1], zmodyfikowanego dla nieholonomicznego charakteru obiektu. W układzie sterowania nadrzędnego oprócz generacji zbioru elementarnych zachowań zrealizowano również ich połączenie z wykorzystaniem sieci neuronowych typu WTA (zwycięzca bierze wszystko) [2, 9].

2. Kinematyka mobilnego robota kołowego

Analizowany mobilny robot pokazany na rysunku 1, jest układem nieholonomicznym o dwóch stopniach swobody. Jako zmienne niezależne opisujące ruch mobilnego robota przyjęto kąty obrotów kół napędzających α_1 i α_2 [3].



Rys. 1. Schemat minirobota.

Z rozkładu wektora prędkości charakterystycznych punktów A, B, C otrzymano:

$$\begin{bmatrix} \dot{\alpha}_1 \\ \dot{\alpha}_2 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} 1 & l_1 \\ 1 & -l_1 \end{bmatrix} \begin{bmatrix} u_v \\ u_\beta \end{bmatrix} \quad (1)$$

gdzie:

$r_1 = r_2 = r$ - promienie kół, u_v - prędkość punktu A, u_β - prędkość kątowna ramy.

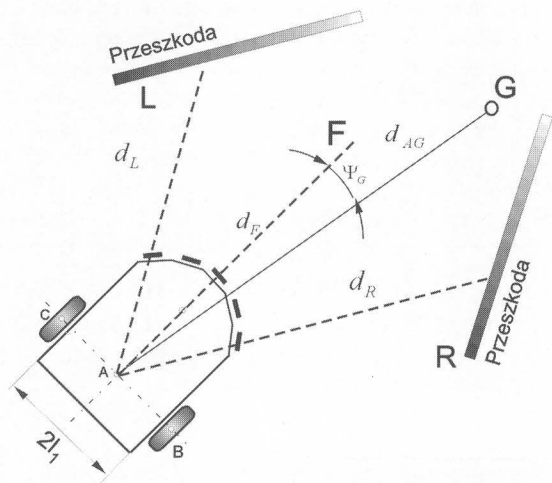
W układzie nieruchomym położenie mobilnego robota opisują współrzędne zdefiniowane jako $[x_A, y_A, \beta]^T$. Kinematykę mobilnego robota kołowego zdefiniowano za pomocą równania [3,4],

$$\begin{bmatrix} \dot{x}_A \\ \dot{y}_A \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} V_{Am} \cos(\beta) & 0 \\ V_{Am} \sin(\beta) & 0 \\ 0 & \omega_m \end{bmatrix} \begin{bmatrix} u_v \\ u_\beta \end{bmatrix} \quad (2)$$

Występujące w zależności (2) wielkości V_{Am} , ω_m są odpowiednio maksymalnymi prędkościami punktu A i ramy mobilnego robota.

3. Sterowanie behawioralne minirobota kołowego

Obiektem, który posłużył do badań poprawności realizacji analizowanych poniżej algorytmów sterowania był mobilny minirobot zbudowany w Katedrze Mechaniki Stosowanej i Robotyki Politechniki Rzeszowskiej [8]. Składa się on z kontrolera Handy Board opartego o procesor 68HC11ERG firmy Motorola, dwóch enkoderów optycznych, modułów napędowych i akumulatorów. Analizowany mobilny minirobot wyposażony jest ponadto w osiem sensorów odległości, umiejscowionych na obwodzie (rysunek 2). Otrzymujemy z nich informacje w postaci odległości od przeszkód d_L, d_F, d_R . Jako wartości d_L, d_F, d_R są przyjmowane minimalne wartości z grupy czujników. Wartość wskazań sensorów jest zdeterminowana przez ich zakres działania i mieści się w przedziale $d_{\min} \leq d_{(.)} \leq d_{\max}$.



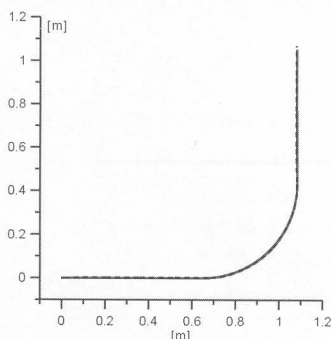
Rys. 2. Schemat minirobota.

W celu uzyskania uniwersalności rozwiązania polegającej na niewrażliwości algorytmu sterowania na rodzaj użytych sensorów wprowadzono normalizację wskazań odległości. Dzięki temu zmiana czujników odległości np. na czujniki o większym zasięgu nie pociąga za sobą konieczności ponownego doboru wartości wag sieci neuronowej dla poprawnej realizacji określonego zachowania.

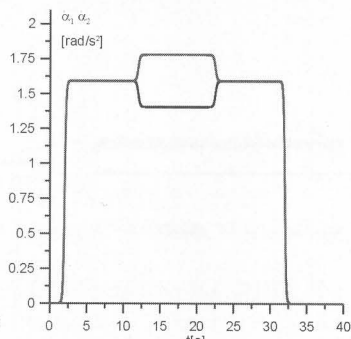
Sygnały z sensorów znormalizowano w następujący sposób: pomiar odległości przez prawy czujnik $d_R^N = d_R / (d_R + d_L)$, przez lewy czujnik $d_L^N = d_L / (d_R + d_L)$ i środkowy $d_F^N = d_F / \rho$, gdzie: ρ -maksymalny zakres pomiarowy sensora.

3.1 Sterowanie niskiego poziomu

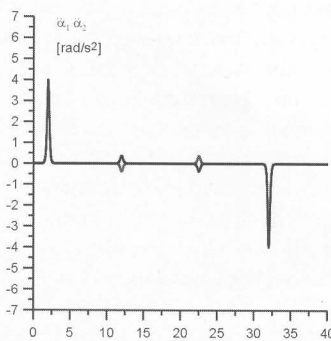
Poprawne wykonywanie zaplanowanego zadania nawigacji w dużym stopniu zależy od dokładności realizacji trajektorii przez układ sterowania niskiego poziomu. W naszym przypadku skorzystano z gotowych algorytmów sterowania mobilnymi robotami, które były przedmiotem wielokrotnych badań i analiz w Katedrze Mechaniki Stosowanej i Robotyki PRz.[3],[4], przyjmując jako układ sterowania niskiego poziomu układ sterowania adaptacyjnego. Po dopasowaniu parametrów układu sterowania do fizycznych parametrów mobilnego minirobota zweryfikowano przyjęte rozwiązanie. Podczas badań eksperymentalnych poprzez dobór parametrów układu minimalizowano błędy realizacji zadanej trajektorii.



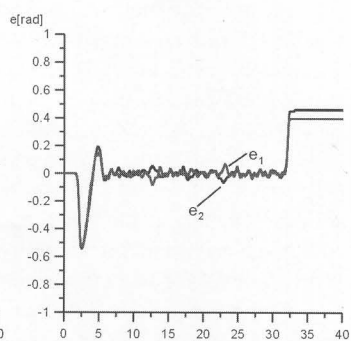
Rys. 3. Tor minirobota



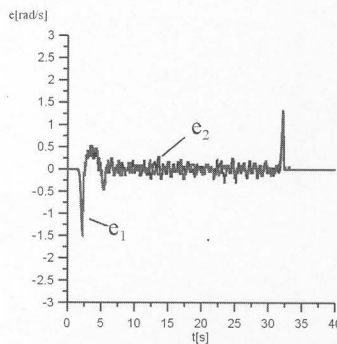
Rys. 4. Prędkości kątowe



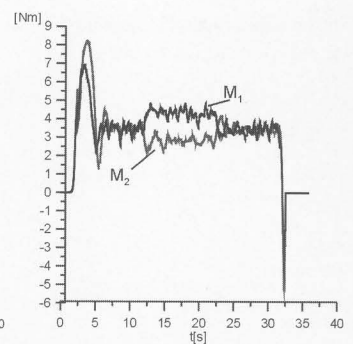
Rys. 5. Przyspieszenia kątowe



Rys. 6. Błędy przemieszczeń kół napędzających.



Rys. 7. Błędy prędkości kół napędzających.



Rys. 8. Sygnały sterujące.

Tor przyjęty do weryfikacji poprawności realizacji układu sterowania pokazano na rysunku 3.

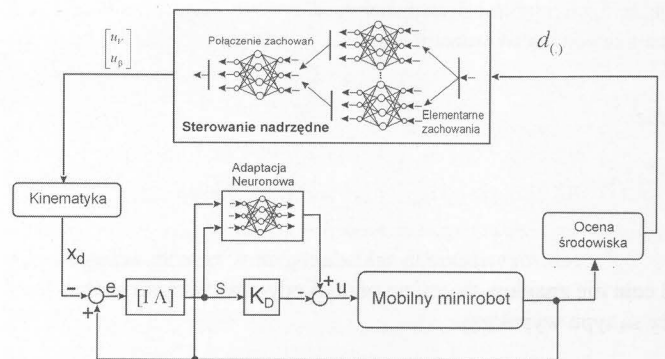
Składa się on z jazdy mobilnego robota po linii prostej, następnie z chwilowej jazdy po torze kołowym i znów poruszanie się po linii prostej.

Uzyskane z badań eksperymentalnych parametry ruchu umożliwiające przemieszczanie się minirobota po zadanej trajektorii zamieszczono na rysunkach 4 i 5. Na rysunku 8 przedstawiono sygnały sterujące. W okresie rozruchu widoczna jest zmiana poziomu sygnału sterowania, podobne zmiany można zauważyć w czasie jazdy po łuku i w okresie hamowania.

Widoczne na rysunku 7 błędy realizacji prędkości oraz przedstawione na rysunku 6 błędy przemieszczeń kół napędzających są ograniczone, największe swoje wartości przyjmują w chwili startu i hamowania.

3.2 Sterowanie nadrzędne

Kolejnym krokiem w realizacji układu neuronowej nawigacji mobilnego minirobota kołowego była synteza i symulacja układu sterowania nadrzędnego pokazanego na rysunku 9. Składa się on z dwu części: pierwszej generującej zbiór elementarnych zachowań oraz drugiej umożliwiającej ich połączenie w celu zrealizowania postawionego zadania nawigacji, np. „idź do celu z omińnięciem przeszkód”.



Rys. 9. Schemat układu nawigacji MRK.

Mając informacje o odległości od przeszkód (rysunek 10) oraz korzystając z równania (1), wektor prędkości kątowych i liniowych $[u_V, u_\beta]^T$ dla realizacji zadania „osiągnij środek wolnej przestrzeni.” zapiszemy w postaci:

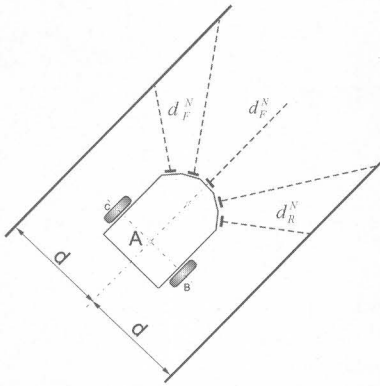
$$\begin{bmatrix} u_{VZSWP} \\ u_{\beta ZSWP} \end{bmatrix} = \frac{r}{2} \begin{bmatrix} 1 & 1 \\ I_1^{-1} & -I_1^{-1} \end{bmatrix} \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \end{bmatrix} \begin{bmatrix} d_L^N \\ d_R^N \\ d_F^N \end{bmatrix} \quad (3)$$

gdzie: $w_{ij} = \text{const}$ to wagi dobranej eksperymentalnie sieci neuronowej.

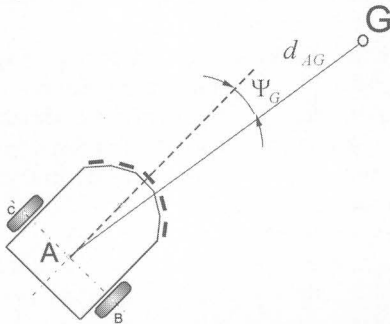
Zachowanie typu „idź do celu” (rysunek 11) realizujemy analogicznie jak omówione powyżej zadanie wyznaczając sygnały $[u_V, u_\beta]^T$ z równania (1)

$$\begin{bmatrix} u_{VCEL} \\ u_{\beta CEL} \end{bmatrix} = \frac{r}{2} \begin{bmatrix} 1 & 1 \\ I_1^{-1} & -I_1^{-1} \end{bmatrix} \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \begin{bmatrix} \psi_G \\ d_{AG} \end{bmatrix} \quad (4)$$

gdzie: Ψ_G - kąt odchylenia robota od celu, d_{AG} - odległość punktu A od punktu docelowego G.



Rys. 10. Schemat zachowania „osiągnij środek wolnej przestrzeni“



Rys. 11. Schemat zachowania „idź do celu“

Realizację zachowania „idź do celu z ominięciem przeszkód“, czyli połączenie dwóch elementarnych zachowań, zrealizowano wykorzystując ideę działania sieci neuronowych typu WTA, gdzie jako kryterium przyjęto minimum prędkości elementarnych zachowań. Sygnały wejściowe pochodzące ze zbioru elementarnych zachowań zdefiniowano jako:

$$u_I = [u_{CEL}, u_{ZSWP}] \quad (5)$$

gdzie: $u_{CEL} = [u_V, u_\beta]^T$, $u_{ZSWP} = [u_V, u_\beta]^T$.

Dla realizacji koordynacji zachowań zaproponowano wygenerowanie sygnału sterowania $u = [u_V, u_\beta]^T$ w postaci:

$$u = u_I W \quad (6)$$

gdzie:

$$W = \begin{cases} 1 & \text{dla } u_{VCEL} \geq \eta \cdot u_{VZSWP} \\ 0 & \text{w innym przypadku} \\ 1 & \text{dla } u_{VCEL} < \eta \cdot u_{VZSWP} \\ 0 & \text{w innym przypadku} \end{cases} \quad (7)$$

wartość współczynnika η ograniczono $0 < \eta < 1$.

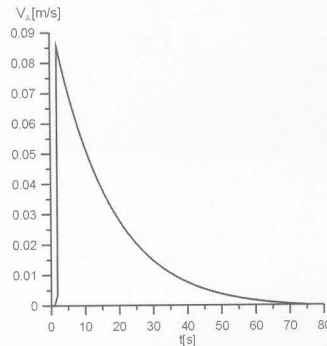
W powyższych rozważaniach zakładamy, że w pewnej odległości od celu nie znajdują się żadne przeszkody oraz wszystkie przeszkody są typu wypukłego.

4. Symulacja i weryfikacja proponowanych rozwiązań

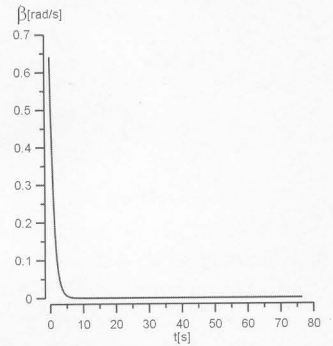
Modelowanie i symulację zaproponowanych rozwiązań przeprowadzono w pakiecie, Matlab/Simulink z wykorzystaniem emulatora przestrzeni roboczej mobilnego robota, zbudowanego w postaci S-funkcji [6,7]. Takie podejście umożliwiło wizualizację trajektorii wybranego punktu mobilnego robota, wyznaczenie odległości od przeszkód, przyjętych w symulacji jako sygnały sensorów oraz dobór wartości wag sieci neuronowej i obserwację wpływu parametrów układu na sposób realizacji określonego zachowania.

Dla potrzeb weryfikacji otrzymanych rozwiązań zaprojektowano stanowisko do badań eksperymentalnych oparte na oprogramowaniu Matlab Simulink, wykorzystując dodatkowo przyborek Real Time Windows Target. Zaproponowane środowisko szybkiego prototypowania, dzięki wykorzystaniu portów LPT komputera umożliwia dwukierunkową transmisję danych pomiędzy sterownikiem, mobilnego robota a zaimplementowanym w pakiecie RTWT algorytmem sterującym.

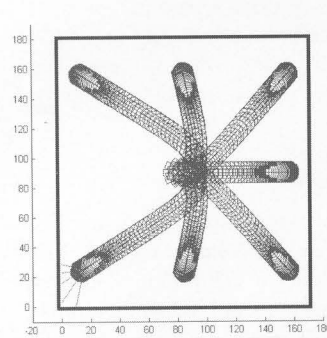
W proponowanym środowisku badawczym zweryfikowano na obiekcie rzeczywistym pracującym w nieznanym środowisku w trybie on-line, wyniki wcześniejszych symulacji elementarnych zachowań. Podczas badań eksperymentalnych rejestrowano błędy realizacji trajektorii wygenerowanej przez nadrzędny układ sterowania, błędy realizacji postawionych elementarnych zachowań: „znajdź środek wolnej przestrzeni“ i „idź do celu“. Dodatkowo kontrolowano poprawność odwzorowania otoczenia przez układ sensoryczny robota, generując z sygnałów sensorów środowisko pracy mobilnego robota i porównując go z rzeczywistym.



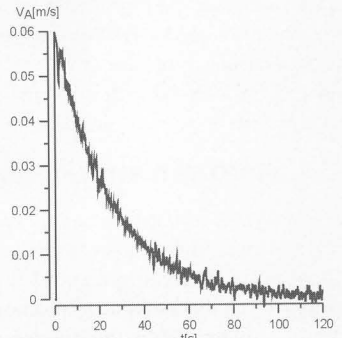
Rys. 12. Prędkość punktu A minirobot



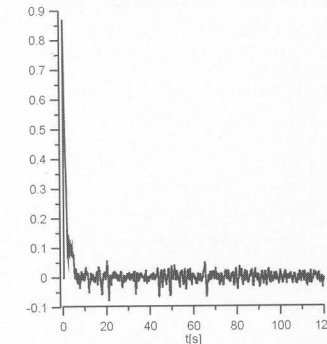
Rys. 13. Prędkość kątowa minirobot.



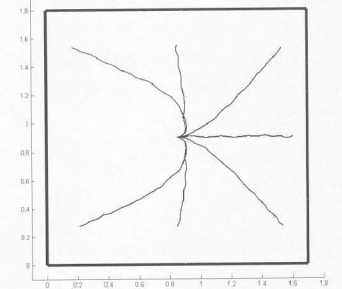
Rys. 14. Realizacja zachowania „idź do celu“.



Rys. 15. Prędkość punktu A minirobot



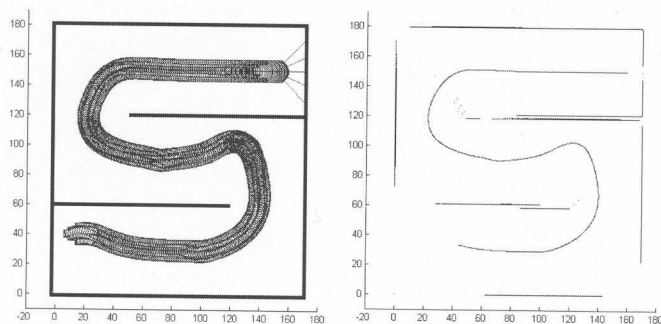
Rys. 16. Prędkość kątowa minirobot.



Rys. 17. Realizacja zachowania „idź do celu“.

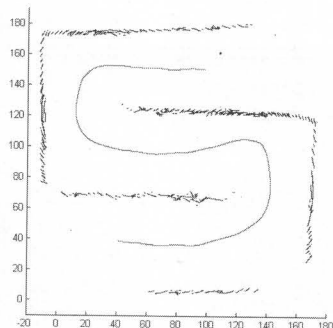
Rysunek 14 przedstawia symulację zadania „Idź do celu“, w którym pokazano osiągnięcie przez mobilny minirobot siedmiu różnych pozycji. Parametry ruchu takie jak: prędkość punktu A miniroboty oraz prędkość kątowa ramy odpowiadające przejazdowi mobilnego miniroboty do celu numer 3 przedstawiają odpowiednio rysunki 12 i 13. Te same parametry ruchu przy realizacji tej samej pozycji, zarejestrowane podczas weryfikacji na obiekcie rzeczywistym, zamieszczono odpowiednio na rysunkach 15 i 16. Na rysunku 17 pokazano zarejestrowane trajektorie charakterystycznego punktu A miniroboty podczas realizacji siedmiu różnych celów przez obiekt rzeczywisty.

Rysunek 18 przedstawia symulację zadania „znajdź środek wolnej przestrzeni“, na którym przedstawiono sposób poruszania się mobilnego miniroboty w nieznanym środowisku a rysunku 19 odtworzone środowisko pracy mobilnego robota korzystając z sygnałów z sensorów oraz wykreślono trajektorię punktu A mobilnego miniroboty. Natomiast na rysunku 20 zamieszczono wyniki eksperymentu realizacji tegoż zachowania.

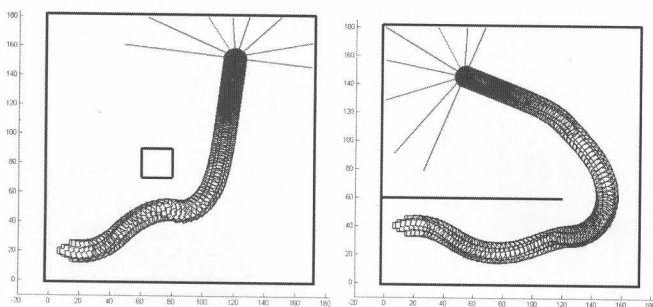


Rys. 18. Realizacja zachowania „znajdź środek wolnej przestrzeni“

Rys. 19. Wygenerowane z symulacji otoczenie minirobota.

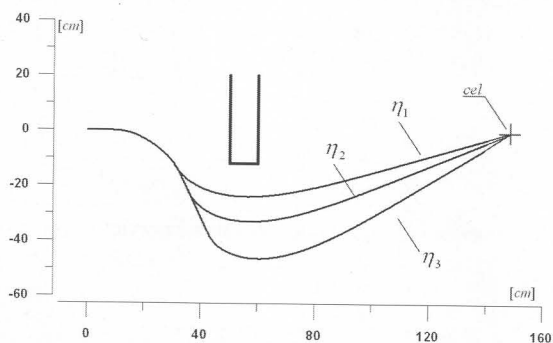


Rys. 20. Eksperymentalna realizacja „znajdź środek wolnej przestrzeni“

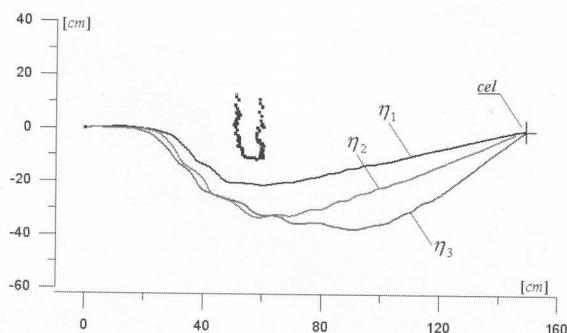


Rys. 21. Realizacja zachowania „idź do celu z omińnięciem przeszkód“

Rys. 22. Realizacja zachowania „idź do celu z omińnięciem przeszkód“



Rys. 23. Symulacja zachowania „idź do celu z omińnięciem przeszkód“.



Rys. 24. Eksperymentalna realizacja zachowania „idź do celu z omińnięciem przeszkód“

Na rysunkach 21 i 22 przedstawiono wyniki symulacji zadania „idź do celu z omińnięciem przeszkód“ zrealizowane dla dwóch różnych celów, w dwóch różnych środowiskach.

Rysunek 23 przedstawia wyniki symulacji zachowania „idź do celu z omińnięciem przeszkód“. Widoczne trajektorie punktu A mobilnego minirobota wygenerowano dla różnych wartości współczynnika η , przy czym $\eta_1 < \eta_2 < \eta_3$. Rysunek 24 prezentuje eksperymentalną realizację wcześniej przedstawionej symulacji.

5. Wnioski

Zamieszczona synteza układu nawigacji mobilnego minirobota kołowego pozwoliła zrealizować dwa elementarne zachowania typu „idź do celu“ oraz „znajdź środek wolnej przestrzeni“. Zbudowany model mobilnego robota i model środowiska, w jakim robot się poruszał pozwoliły na weryfikację numeryczną przyjętego rozwiązania. Otrzymane wyniki symulacji elementarnych zachowań, zweryfikowano na obiekcie rzeczywistym w zaprojektowanym środowisku badawczym. Uzyskane wyniki ich realizacji można uznać za zadowalające.

W pracy przeprowadzono również badania symulacyjne i weryfikację koordynacji elementarnych zachowań z wykorzystaniem sieci neuronowych typu WTA gdzie jako kryterium przyjęto minimum prędkości punktu A podczas realizacji elementarnych zachowań.

Literatura

- [1] Arkin R.C. Behavior-Based Robotics, The MIT Press, 1998.
- [2] Fagg A.H., Lotspeich D., Hoff J., Bekey G.A.. Rapid Reinforcement Learning for Reactive Control Policy Design in Autonomous Robots, Submitted to WCNN 1994.
- [3] Giergiel J., Hendzel Z., Żylski W.: „Kinematyka, dynamika i sterowanie mobilnych robotów kołowych w ujęciu mechatronicznym“, Monografie, Wyzd.ImiR, AGH Kraków 2000
- [4] Giergiel J., Hendzel Z., Żylski W., Kinematyka, dynamika i sterowanie mobilnych robotów kołowych, PWN Warszawa 2002.
- [5] Hendzel Z, Neural Network Reactive Navigation and Control of Wheeled Mobile Robot, Advances in Soft Computing, Rutkowski, Kacprzyk, Eds., Phisica-Verlag, 2003 str.686-691.
- [6] Hendzel Z., Burghardt A., Neuronowy emulator mobilnego robota kołowego, Zeszyty Naukowe Politechniki Rzeszowskiej, Mechanika z.60, Problemy dynamiki konstrukcji, Oficyna Wydawnicza Politechniki Rzeszowskiej, str. 242-248.
- [7] Hendzel Z., Burghardt A., Prototypowanie neuronowego emulatora mobilnego robota kołowego, Projektowanie Mechatroniczne, praca zbior. pod red. T.Uhla Wydawnictwo Katedry Robotyki i Dynamiki Maszyn AGH Kraków 2002, str. 72-79.
- [8] Hendzel Z., Giergiel J., Burghardt A., Neuronowa nawigacja odruchowa i sterowanie mobilnego minirobota, Projektowanie Mechatroniczne Zagadnienia Wybrane, praca zbior. pod red. T.Uhla Wydawnictwo Katedry Robotyki i Dynamiki Maszyn AGH Kraków 2003, str. 59-68.
- [9] Tadeusiewicz R., Elementarne wprowadzenie do techniki sieci neuronowych z przykładami, Problemy współczesnej nauki teoria i zastosowania, Informatyka Akademicka Oficyna Wydawnicza PLJ Warszawa 1998
- [10] Zhang Y. L., Velinsky S.A., Feng X. On the Tracking Control of Differentially Steered Wheeled Mobile Robots, Journal of Dynamic Systems, Measurement, and Control, September 1997, Vol. 119(3): 455 - 461.

Praca została zrealizowana w ramach Grantu KBN, nr 5 T07C 007 22

Title: Behavioral control of a mobile mini robot

Artykuł recenzowany