

Mariusz DĄBKOWSKI

POLITECHNIKA GDAŃSKA, WYDZIAŁ ELEKTROTECHNIKI I AUTOMATYKI, KATEDRA ROBOTYKI I SYSTEMÓW MECHTRONIKI
ul. Własna Strzecha 18a, 80-233 Gdańsk

System nawigacji bezkolizyjnej autonomicznego robota mobilnego SCORPION

Dr inż. Mariusz Dąbkowski

Jest adiunktem w Katedrze Robotyki i Systemów Mechatroniki na Wydziale Elektrotechniki i Automatyki Politechniki Gdańskiej. Studia na tym samym wydziale ukończył w 2002 r. Stopień doktora nauk technicznych uzyskał w 2006 r. Przedmiotem jego zainteresowań naukowych jest robotyka, a w szczególności robotyka mobilna.



e-mail: m.dabkowski@ely.pg.gda.pl

Streszczenie

W artykule przedstawiono opracowany i wykonany system nawigacji autonomicznego robota mobilnego ERSP SCORPION umożliwiającą realizację zadania unikania kolizji z przeszkodami statycznymi znajdującymi się w otoczeniu, w którym robot się porusza. Do akwizycji wiedzy o otoczeniu wykorzystano czujniki odometryczne oraz podczerwieni aktywnej. Przedstawiono również wyniki badań systemu nawigacji w różnych warunkach pracy, począwszy od braku przeszkód statycznych, poprzez środowisko z pojedynczymi przeszkodami, a kończąc na złożonych układach, z większą liczbą obiektów w otoczeniu robota.

Słowa kluczowe: roboty mobilne, nawigacja robotów, systemy bezkolizyjne, budowa map otoczenia.

Collision-free navigation system for autonomous mobile robot SCORPION

Abstract

The paper presents a developed and realized system of autonomous navigation for mobile robot ERSP SCORPION. It enables the collision avoidance with static obstacles located in the robot's environment. New in the navigation algorithm is applying additional behavioral mechanisms, which helped to minimize, or in some cases, to eliminate sensory errors (IR and rotary-pulse), which reducing the quantity of executed task. The diagram of the developed motion algorithm is illustrated in Fig. 3. Series of dozens of real navigation tests by robot SCORPION, for different configurations of obstacles in the environment, were performed, to examine the developed system. In paper, exemplary results of navigation test for different collision scenarios, are presented. The motion path without obstacles in environment is shown in Fig. 4. In Fig. 6 the path, for a single obstacle, is demonstrated. Location of obstacle is shown in Fig. 5. In Fig. 8 the navigation path for many obstacles (schema according to the diagram on Fig. 7), is presented. The real path of robot ERSP SCORPION is demonstrated in Fig. 9. Errors of achieving of goal position by the robot are shown in Tab. 1. Obtained values confirm increasing the odometry errors with the traveled distance. Results of verification tests indicated that the proposed algorithm works correctly and the robot SCORPION avoids collisions each time.

Keywords: mobile robots, mobile robot's navigation, collision avoidance, map building.

1. Wprowadzenie

Systemy nawigacji bezkolizyjnej autonomicznych robotów mobilnych stanowią jedną z najsilniej i najdynamiczniej rozwijających się gałęzi wiedzy w dziedzinie robotyki mobilnej. Podstawowym aspektem bezkolizyjności jest umiejętność poruszania się robota w środowisku pracy bez fizycznego kontaktu z przeszkodami, a osiągnięcie tego stanu wiąże się z akwizycją danych z czujników pomiarowych określających przemieszczenie, jak również z czujników wykrywających obecność pobliskich obiektów. Dodatkowo, w celu zapewnienia wysokiej efektywności

działania systemu nawigacji, pozyskiwane dane – z pokładowych sensorów platformy mobilnej – muszą cechować się wysoką dokładnością. Uzyskanie takiego stanu zapewnia m.in. odpowiednia liczba, rozmieszczenie i jakość czujników robota, przy czym całość realizowanych funkcji jest nadzorowana przez odpowiedni algorytm logiczny, który jest implementowany w formie kodu programu.

W artykule przedstawiono opracowany i wykonany system nawigacji autonomicznego robota mobilnego ERSP SCORPION, w którym do unikania kolizji z przeszkodami nieruchomymi wykorzystano czujniki obrotowo-impulsowe oraz podczerwieni aktywnej. Przedstawiono również wyniki badań opracowanego systemu w różnych warunkach pracy, począwszy od braku przeszkód statycznych, poprzez środowisko, w którym występują pojedyncze przeszkody, a skończywszy na złożonych układach, z większą liczbą obiektów w otoczeniu robota.

2. Metody nawigacji robotów mobilnych

W nowoczesnych systemach nawigacji robotów mobilnych wykorzystuje się rozmaite metody opisywane w bogatej literaturze przedmiotu, m.in. [1, 2, 3, 4, 5, 7, 8, 9, 10]. Do najpopularniejszych spośród występujących w tych opisach rozwiązań należy zaliczyć: dead reckoning, aktywne wieże, znaczniki oraz mapy otoczenia.

Dead reckoning jest rodzajem nawigacji polegającym na określeniu aktualnego położenia oraz przemieszczenia robota wyłącznie za pomocą jego czujników pokładowych, takich jak enkodery obrotowo-impulsowe. Wadą tej metody jest kumulująca się wartość błędu określenia położenia robota wraz z przejechanym dystansem. Stąd też najczęściej metoda ta jest stosowana w połączeniu z innymi sposobami definiowania pozycji lub też z innymi czujnikami, np.: zbliżeniowymi. Pozwalają one nie tylko wykryć przeszkody w otoczeniu robota, ale również umożliwiają określenie z większą dokładnością położenie robota, co może być wykorzystane do utworzenia mapy otoczenia, która pozwoli na precyzyjne planowanie ruchu robota.

Aktywne wieże znajdują szczególnie szerokie zastosowanie w ruchu lotniczym i morskim, w przypadku zastosowania których, precyzyjne pozycjonowanie obiektu nie wymaga skomplikowanych i czasochłonnych obliczeń, ale wiąże się ze znacznymi kosztami budowy i utrzymania całej infrastruktury. Nawigacja w tym przypadku bazuje na dwóch powszechnie stosowanych metodach estymacji pozycji obiektu, tj.: trilateracji i triangulacji [1].

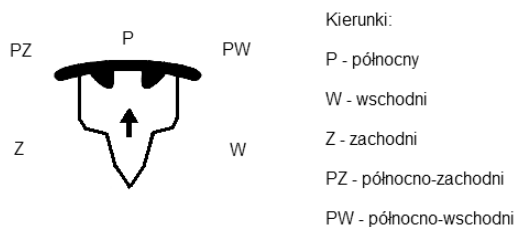
Znaczniki to z kolei obiekty wyróżniające się z otoczenia, które robot może z łatwością rozpoznać za pomocą pokładowych czujników. Mogą one mieć rozmaite kształty, które wykrywając, np.: za pomocą dedykowanych do tego celu kamer wizyjnych, system nawigacji wykorzystuje do orientacji na powierzchni terenu. Znaczniki dzielą się na dwie grupy: naturalne – czyli takie które w otoczeniu pracy robota istnieją, ale nie zostały stworzone w celu jego nawigacji oraz sztuczne – projektowane specjalnie do celów związanych z nawigacją robotów mobilnych.

Mapy otoczenia stanowią wzorce do porównywania, dzięki temu pozwalają na szybszą orientację robota w terenie oraz na efektywniejsze działanie systemu nawigacji. Takie systemy mają wiele zalet, gdyż pozwalają nie tylko na wykorzystywanie naturalnego otoczenia do planowania trasy robota, ale także dają możliwość „nauczenia się nowych terenów” dzięki czemu możliwe staje się m.in. usprawnienie dokładności jego pozycjonowania. Istnieją jednak w tym przypadku pewne dodatkowe wymagania, które należy spełnić. Przede wszystkim przeszkody muszą być statyczne i łatwo rozpoznawalne dla sensorów robota, tzn. o wymiarach pozwalających na łatwą ich identyfikację, np. przedmioty wąskie i długie. Przedmioty o powierzchniach przezroczystych lub

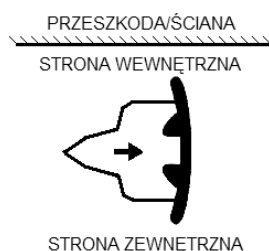
odblaskowych mogą nie zostać właściwie odnotowane przez czujniki. Ponadto mapa otoczenia musi być szczegółowa, co z kolei jest procesem czasochłonnym z punktu widzenia platformy robota i wiąże się ze zużyciem znacznych ilości energii [1].

3. Algorytmy w systemie nawigacji bezkolizyjnej robota SCORPIO

W celu realizacji głównego zadania nawigacji bezkolizyjnej w otoczeniu z przeszkodami nieruchomymi wykorzystano odpowiednie nazewnictwo kierunków ruchu robota, którego znaczenie zilustrowano na rys. 1 i rys. 2.



Rys. 1. Nazewnictwo kierunków ruchu robota
 Fig. 1. Denotations of motion directions of the mobile robot



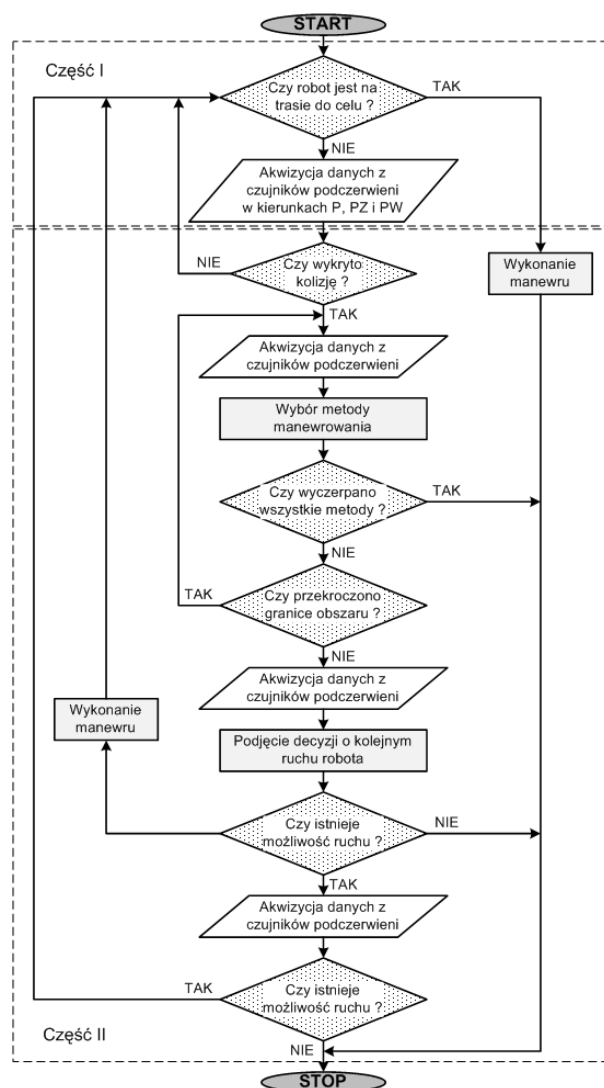
Rys. 2. Nazewnictwo usytuowania robota na powierzchni
 Fig. 2. Position of the robot in environment and names of left and right zone

Na rys. 3 przedstawiono opracowany i zaimplementowany w systemie sterowania algorytm nawigacji bezkolizyjnej robota ERSP SCORPIO. Dzieli się on na dwie części, odpowiedzialne za ruch postępowy robota (część I) oraz za omijanie napotkanej przeszkody (część II).

Informacje o aktualnym położeniu robota, które są wykorzystywane przez algorytm nawigacji, są dostarczane z odpowiedniego układu odometrycznego. Jeżeli robot znajduje się w odległości do 25 cm od celu, wówczas wykonuje on ruch w kierunku miejsca docelowego i kończy działanie, w przeciwnym wypadku przemieszcza się o 25 cm do przodu i wykonuje pomiar czujnikami podczerwieni w kierunkach: P, PZ i PW. Cały cykl ruchu postępowego jest wykonywany w pętli, aż do chwili napotkania przeszkody na trasie platformy mobilnej. W takiej sytuacji jest określana (i w razie potrzeby korygowana) odległość od przeszkody, po czym na podstawie danych pochodzących z kolejnego pomiaru za pomocą czujników podczerwieni podejmowana jest decyzja o sposobie objazdu przeszkody i następuje przejście do drugiej części algorytmu [6].

Część algorytmu odpowiedzialnego za omijanie przeszkód składa się z trzech etapów objazdu przeszkody, po jednym na każdą ze stron, wzdłuż których robot będzie się poruszał. Każdy z etapów zakłada istnienie przeszkody, która całkowicie zachodzi na tor ruchu robota i wymaga oddzielnego manewru omijania. Główną funkcją tej części algorytmu, wykonywaną po każdym ruchu robota, jest sprawdzenie czy nie przekroczył on określonych programowo granic terenu. W razie wystąpienia takiej sytuacji, robot powraca do pozycji wyjściowej manewru omijania i inicjuje objazd przeszkody drugą stroną lub – w przypadku nieudanych obydwu prób – kończy działanie. Natomiast w przypadku, gdy granice nie zostały przekroczone, wykonywany jest kolejny pomiar czujnikami podczerwieni i podejmowana jest decyzja

o kolejnym ruchu robota. Przy czym, w zależności od wartości danych pomiarowych robot inicjuje omijanie przeszkody, koryguje odległość od obecnego obiektu i powraca do pozycji wyjściowej, a w przypadku braku zagrożeń – wykonuje ruch wzdłuż ściany przeszkody. Następnie, jest wykonywany kolejny pomiar czujnikami podczerwieni i następuje weryfikacja obecności omijanej przeszkody. W zależności od uzyskanych danych robot inicjuje kolejną pętlę cyklu i przechodzi do kolejnego etapu objazdu lub kończy cały manewr i ponownie rozpoczyna ruch postępowy w stronę punktu docelowego [6].



Rys. 3. Schemat algorytmu nawigacji bezkolizyjnej autonomicznego robota mobilnego SCORPIO

Fig. 3. The algorithm scheme of the collision avoidance for autonomous mobile robot SCORPIO

Algorytm zaprojektowano w ten sposób, aby domyślnym, preferowanym kierunkiem omijania przeszkody był lewa strona robota. Jeśli konfiguracja obiektów uniemożliwi ten manewr lub kiedy się on nie powiedzie, robot realizuje proces okrążenia przeszkody od strony prawej.

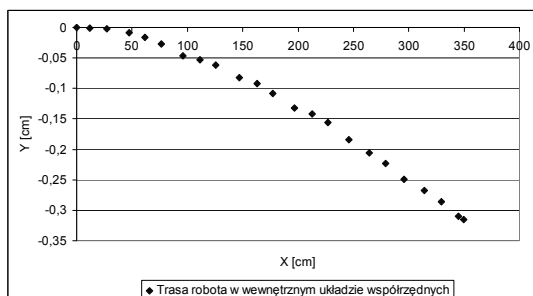
W celu zminimalizowania niepożądanego procesu utykania robota w minimach lokalnych uwzględniono w algorytmie zachowanie omijania przeszkody dodatkowej znajdującej się na torze kolizyjnym, podczas manewru okrążania pierwszego napotkanego na drodze obiektu. Dzięki temu robot z powodzeniem wydostaje się z zaułków w kształcie litery "u". Niestety wykrycie następnej przeszkody podczas manewru omijania przeszkody dodatkowej skutkuje jego przerwaniem i powrotem do pozycji wyjściowej, a następnie rozpoczęciem próby okrążania obiektu drugą stroną lub zakończeniem misji (jeśli manewr omijania w drugą stronę

miał już miejsce). Ponadto robot wróci do pozycji wyjściowej, jeśli osiągnie zdefiniowaną, w wewnętrznym układzie współrzędnych, wirtualną granicę obszaru.

Opracowany algorytm do bezkolizyjnej nawigacji nie uwzględnia ograniczeń nieholonomicznych platformy ze względu na to, że do nawigacji bezkolizyjnej używane są praktycznie sensory podczerwieni aktywnej, natomiast czujniki obrotowo-impulsowe jedynie do określania położenia robota w wirtualnym układzie współrzędnych. Zatem nawet, gdyby wystąpiły poślizgi kół, to algorytm wykorzystując tylko odczyty odległości jest w stanie realizować zadanie unikania kolizji poprawnie. Ponadto robot ERSP SCORPION wyposażony jest system łagodnego startu silników napędowych kół, co w połączeniu z dużym współczynnikiem tarcia występującym na styku koło-podłoże (guma-linoleum), praktycznie eliminuje poślizgi.

4. Weryfikacja działania opracowanego systemu nawigacji robota

Prowadząc weryfikację działania opracowanego systemu, wykonano serię kilkudziesięciu rzeczywistych badań, dla różnych konfiguracji przeszkód w środowisku pracy robota. Należy dodać, że poszczególne wartości współrzędnych położenia robota na powierzchni terenu, określane przez system odometryczny wspomagany czujnikami IR, wyrażone były w wewnętrznym układzie współrzędnych platformy zaczepionym w środku robota, gdzie oś X pokrywa się z osią symetrii wzdłużnej platformy. Ze względu na ograniczoną ilość miejsca w artykule przedstawiono trzy przykładowe trasy robota. Wyniki badań, dla przypadku najprostszego – przy braku przeszkód – przedstawiono na rys. 4 [6].

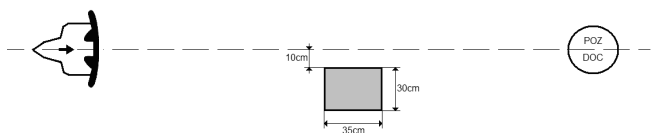


Rys. 4. Zarejestrowany przebieg trasy robota w przypadku braku przeszkód (scenariusz 0) [6]

Fig. 4. The path of the robot without obstacles in environment (scenario 0)

Na drodze o długości 3,5 m zarejestrowane odchylenie pozycji, po zakończeniu działania robota, wynosiło zaledwie 3,15 mm, co w rozpatrywanym przypadku można uznać za wynik w pełni satysfakcjonujący [6].

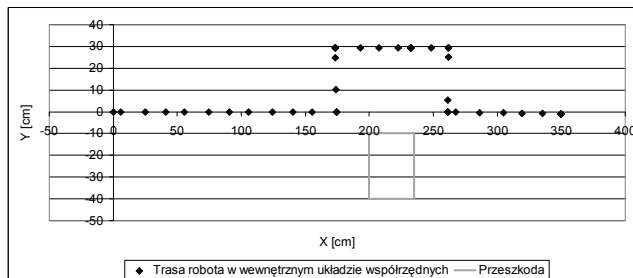
W przypadku występowania przeszkód pojedynczych, najtrudniejszym okazał się przypadek, w którym przeszkoda zachodzi na tor ruchu robota, stwarzając ryzyko kolizji i utrudniając poprawne jej wykrycie (rys. 5).



Rys. 5. Przeszkoda nie zachodząca na linię osi symetrii robota, i do niej prostopadła (scenariusz 1) [6]

Fig. 5. The obstacle is not occurring on the axis of symmetry of the robot and is perpendicular to it (scenario 1)

Uzyskane wyniki badań, dla tego przypadku (rys. 6), świadczą jednak o prawidłowym zachowaniu się systemu nawigacji robota. System nie tylko bezbłędnie określił, z której strony znajduje się przeszkoda (i na tej podstawie obrał lewostronną trasę jej omijania), ale również precyzyjnie i bezkolizyjnie doprowadził układ robota do punktu docelowego.

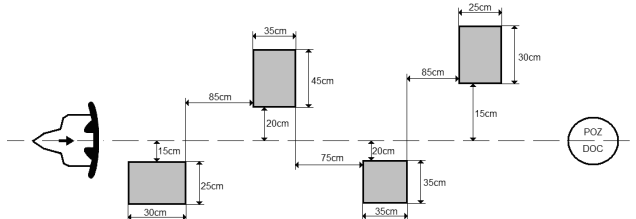


Rys. 6. Zarejestrowany przebieg trasy robota w obecności pojedynczej przeszkody nie zachodzącej na linię osi symetrii robota i ułożonej do niej prostopadła (scenariusz 1)

Fig. 6. The path of the robot with a single obstacle which is not occurring on the symmetry axis of the robot and is perpendicular to it (scenario 1)

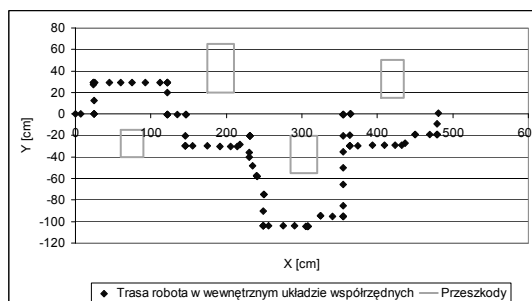
W przypadku występowania wielu przeszkód, jednym z najtrudniejszych okazał się przypadek, w którym przeszkody były rozmieszczone wzdłuż osi X w stosunku do wewnętrznego układu odniesienia robota. Z założenia robot miał ominąć przeszkody pierwszą i trzecią lewostronnie, a pozostałe dwie prawostronnie (rys. 7).

Przeprowadzony w tym przypadku szereg przejazdów testowych wykazał, że uzyskane trajektorie ruchu różniły się od siebie, w zależności od ustawienia początkowego robota oraz od błędów pomiarów z czujników podczerwieni. W części przypadków trajektoria ruchu była zgodna z założeniem, a w innych w sposób istotny odbiegała od zamierzeń, co jednakże nie zmieniało faktu, iż za każdym razem punkt docelowy był osiągnięty bezkolizyjnie (rys. 8 i 9) [6].



Rys. 7. Układ wielu przeszkód znajdujących się na trasie robota (scenariusz 2) [6]

Fig. 7. Configuration of many obstacles along the path of the robot (scenario 2)



Rys. 8. Zarejestrowany przebieg trasy robota w środowisku z wieloma przeszkodami (scenariusz 2) [6]

Fig. 8. The path of the robot with many obstacles in environment (scenario 2)

Na rys. 9 przedstawiono przykładową, rzeczywistą ścieżkę ruchu robota (jego punktu charakterystycznego – środka platformy) dla scenariusza 2.

Estymacja położenia robota na płaszczyźnie przeprowadzana była, jak już wspomniano, w wewnętrznym, wirtualnym układzie współrzędnych robota. W rzeczywistości określenie pozycji za pomocą odometrii jest obciążone błędem, który wzrasta wraz z długością przebytej drogi, stąd zaszła konieczność pomiarzenia prawdziwej pozycji robota, zwłaszcza punktu docelowego. W tym celu przeprowadzono szereg przejazdów testowych robota dla opisanych powyżej scenariuszy, a wyniki błędów osiągnięcia położenia zadanego ($x_d=3500$ mm, $y_d=0$ mm – scenariusz 0 i 1 oraz

$x_d=4800$ mm, $y_d=0$ mm – scenariusz 2) robota oraz obliczone wartości podstawowych charakterystycznych wielkości statystycznych przedstawiono w tabeli 1 [11].



Rys. 9. Rzeczywista ścieżka ruchu robota ERSP SCORPION (scenariusz 2)
Fig. 9. The real path of robot ERSP SCORPION (scenario 2)

Tab. 1. Błędy położenia robota dla różnych scenariuszy kolizyjnych
Tab. 1. Positioning errors of the robot in various collision scenarios

| Lp. | Scenariusz 0 | | Scenariusz 1 | | Scenariusz 2 | | |
|---|--------------------|--------|--------------|--------|--------------|--------|---------|
| | $ Δx $ | $ Δy $ | $ Δx $ | $ Δy $ | $ Δx $ | $ Δy $ | |
| | [mm] | [mm] | [mm] | [mm] | [mm] | [mm] | |
| 1 | 5 | 26 | 30 | 38 | 238 | 95 | |
| 2 | 9 | 34 | 10 | 73 | 5 | 21 | |
| 3 | 7 | 6 | 12 | 89 | 131 | 133 | |
| 4 | 6 | 40 | 3 | 45 | 50 | 242 | |
| 5 | 9 | 12 | 5 | 68 | 63 | 82 | |
| 6 | 7 | 28 | 19 | 3 | 53 | 34 | |
| 7 | 6 | 8 | 9 | 55 | 109 | 219 | |
| 8 | 5 | 30 | 13 | 4 | 92 | 464 | |
| 9 | 2 | 8 | 39 | 2 | 9 | 20 | |
| 10 | 10 | 11 | 3 | 9 | 130 | 282 | |
| Wartość średnia $ Δx $ i $ Δy $ | [mm] | 6,6 | 20,3 | 14,3 | 38,6 | 88,0 | 159,2 |
| Wariancja S^2 $ Δx $ i $ Δy $ | [mm ²] | 5,6 | 158,2 | 141,6 | 1064,3 | 4777,1 | 20370,4 |
| Odchylenie standardowe S $ Δx $ i $ Δy $ | [mm] | 2,4 | 12,6 | 11,9 | 32,6 | 69,1 | 142,7 |

Wszystkie pomiary zostały dokonane za pomocą taśmy mierniczej o dokładności ± 1 mm. Wartość średnia modułu błędu bezwzględnego $|Δy|$ dla scenariusza 0 wyniosła 20,3 mm, czyli była ponad sześciokrotnie większa od błędu określenia pozycji przez system odometryczny (3,15 mm). Dla scenariusza 1 błąd odometrii $|Δy|$ wyniósł 9,6 mm i był przeszło czterokrotnie mniejszy niż wartość rzeczywista. Natomiast wartość $|Δy|$ dla scenariusza 2 była największa spośród wszystkich przedstawionych i wyniosła 159,2 mm, czyli około ośmiokrotnie więcej niż wartość położenia estymowanego za pomocą odometrii (20,2 mm). W przypadku osiągnięcia pozycji w osi X różnice między wartościami błędów osiągnięcia zadanej pozycji w rzeczywistości i w układzie wewnętrznym były podobne, a tendencja wzrostowa wraz z wydłużeniem się przebytej drogi jest wyraźnie widoczna. Uzyskane wyniki są zgodne z oczekiwaniami. Tak duże różnice pomiędzy analogicznymi wartościami położenia (odometria – taśma miernicza) biorą się stąd, że odometria nie uwzględnia takich czynników, jak: różnice w średnicach kół, niesymetrię zamocowania kół względem osi robota, czy nierówności podłoża. Zadaniem zaprojektowanego algorytmu sterowania było jednakże bezkolizyjne prowadzenie robota w otoczeniu przeszkód nieruchomych, co w głównej mierze realizowane było na podstawie odczytów z czujników podczerwieni, stąd też nawet duże błędy odometryczne nie wpłynęły na powodzenie misji.

Testy weryfikacyjne wykazały, że więzy nieholonomiczne ruchu robota nie zostały w żadnym miejscu zerwane. Platforma poruszała się bez poślizgu, co zapewnił system łagodnego startu silników napędowych kół oraz duży współczynnik tarcia między kołami a podłożem (guma–linoleum).

5. Podsumowanie i wnioski

Podsumowując opisane w artykule, wybrane rozważania można stwierdzić, że proponowany algorytm nawigacji bezkolizyjnej działa w pełni poprawnie. Świadczą o tym przeprowadzone wyniki badań, w przypadku których działanie algorytmu kończyło się osiągnięciem punktu docelowego w sposób całkowicie bezkolizyjny. Wprawdzie zdarzały się przypadki wyboru bardziej złożonej (aniżeli najprostszą dla danego przypadku) trajektorii ruchu, lecz w żadnym z przebadanych przypadków nie wpływało to na rezultat końcowy, którym było osiągnięcie przez system pozycji docelowej.

Pewne niedogodności w działaniu systemu stwarzała niedostateczna liczba czujników podczerwieni po obu stronach robota – po jednym czujniku na każdą ze stron. Ponadto czujniki zamontowane były na głębokości 14 cm od bocznej granicy platformy, w wyniku czego robot często, po przejściu do drugiego etapu omijania, prawie natychmiast obracał się w stronę przeszkody. Było to spowodowane faktem, że boczny czujnik podczerwieni jeszcze nie znalazł się na wysokości przeszkody, co powodowało wykrywanie wolnej przestrzeni, którą można się dostać na tor główny. Stąd też implementacja algorytmu wymagała dodatkowych zabezpieczeń weryfikacji zajętości przestrzeni roboczej po obrocie, a to z kolei wiązało się wydłużeniem czasu przejazdu oraz ze zwiększonym zużyciem energii.

Innym ważnym problemem wynikającym z takiej konfiguracji czujników podczerwieni była sytuacja, w której robot omijając przeszkodę zatrzymywał się w celu sprawdzenia wskazań czujników, a boczny czujnik widział w tym czasie końcowy odcinek przeszkody. Efektem tego było inicjowanie kolejnego przesunięcia robota (o 20 cm), po którym dopiero wykrywany był koniec aktualnej przeszkody.

6. Literatura

- [1] Borenstein J., Everett H. R., Feng L.: Where Am I? Sensors and Methods for Mobile Robot Positioning. University of Michigan, USA, 1996 r.
- [2] Spong M. W., Vidyasagar M.: Dynamika i sterowanie robotów. WNT, Warszawa 1997.
- [3] Giergiel M. J., Hendzel Z., Żylski W.: Modelowanie i sterowanie mobilnych robotów kołowych. Warszawa, Wydawnictwa Naukowe PWN 2002.
- [4] Arkin R. C.: Behavior-based robotics. MIT press. MA. 1998.
- [5] Woźniak A.: Autonomiczne roboty mobilne – laboratorium. Poznań: Wydawnictwo Politechniki Poznańskiej 1994.
- [6] Pazderski A.: Nawigacja autonomicznego robota mobilnego ERSP SCORPION z wykorzystaniem czujników podczerwieni aktywnej. Praca dyplomowa. Politechnika Gdańska, Gdańsk 2009.
- [7] Dulęba I.: Metody i algorytmy planowania ruchu robotów mobilnych i manipulacyjnych. Warszawa, Akademicka Oficyna Wydawnicza EXIT 2001.
- [8] MacKenzie D. C., Balch T. R.: Making a clean sweep – behavior based vacuuming. In Proceedings of the AAAI Fall Symposium, Instatationing Real-World Agents.1996.
- [9] Neumann de Carvalho R., Vidal H. A., Vieira P., Ribeiro, M.I.: Complete coverage path planning and guidance for cleaning robots. In Proceedings of the IEEE International Symposium on Industrial Electronics. 1997.
- [10] Acar E. U., Choset H., Atkar P. X.: Complete sensor-based coverage with extended-range detectors – a hierarchical decomposition in terms of critical points and Voronoi diagrams. In: Proceedings of the IEEE International Conference on Intelligent Robots and Systems. 2001.
- [11] Oktaba W.: Elementy statystyki matematycznej i metodyka doświadczalnictwa. Warszawa: Państwowe Wydawnictwa Naukowe 1977.