

Robot mobilny o zmiennym sposobie lokomocji – system sterowania

Tomasz Winiarski, Dawid Seredyński

Instytut Automatyki i Informatyki Stosowanej, Politechnika Warszawska

Streszczenie: W artykule przedstawiono opis systemu sterowania robota mobilnego o dwóch współosiowych kołach, mogącego poruszać się w dwóch trybach lokomocji: dynamicznie stabilnym oraz statycznie stabilnym. Robot może zmieniać tryb ruchu przez automatyczny manewr wstawiania do pionu. System sterowania stanowią kaskady regulatorów PID.

Słowa kluczowe: robot mobilny, odwrócone wahadło, kaskada PID

Wspólną cechą większości zrobotyzowanych platform mobilnych jest napęd różnicowy, którego podstawowymi zaletami są prosta realizacja i zapewnienie wystarczającej manewrowości robota w dedykowanych zastosowaniach. Oparte na nim roboty saperskie o napędzie gąsienicowym [1] wspierają człowieka w niebezpiecznych zadaniach (np. w obszarze zagrożonym wybuchem ładunków albo działaniem terrorystów). Przemieszczają się w trudnym terenie, mogą przepychać przeszkody i są stabilną bazą dla manipulatorów. Z drugiej strony, ich ruch pochłania znaczne ilości energii, dodatkowo ze względu na poślizgi samolokalizacja robota jest utrudniona, gdyż przy takiej konfiguracji informacja z enkoderów zamocowanych na silnikach jest zdecydowanie niewystarczająca.

Wychodząc na przeciw powyższym problemom, stworzono całą gamę robotów o dwukołowym napędzie różnicowym, które przy odpowiedniej konstrukcji mogą poruszać się w trybie odwróconego wahadła. Tego typu roboty wprawdzie nie posiadają takich dobrych właściwości do przemieszczania się w terenie w obecności przeszkód jak roboty saperskie, ale za to w praktycznych realizacjach z powodzeniem przewożą ludzi [2], zużywając przy tym relatywnie mało energii. Co więcej, stojąc w pionie w swojej nominalnej pozycji, zajmują mało miejsca na powierzchni, co pozwala im na poruszanie się w ograniczonej przestrzeni. Na bazie odwróconego wahadła budowano też roboty wyposażone w ręce zdolne do werbalnej i niewerbalnej komunikacji z ludźmi [3]. Roboty mobilne działające na zasadzie odwróconego wahadła są także z powodzeniem wykorzystywane jako manipulatory humanoidalne [4] oraz pojazdy transportowe [5].

Sterowanie balansującym robotem mobilnym wymaga, ze względu na niestabilną naturę, szybkiej odpowiedzi układu sterowania na zmiany orientacji i stanu robota. Konieczne jest zapewnienie dokładnego i pozbawionego opóźnień pomiaru kąta odchylenia robota od pionu [6]. Do pomiaru kąta pochylenia robota wykorzystuje się zwykle pomiary z kilku czujników, filtrowane za pomocą filtru Kalmana [7] lub filtru komplementarnego [8]. Z drugiej

strony konieczna jest synteza odpowiedniego układu regulacji. W licznych pracach zastosowano różne rozwiązania, m.in. logikę rozmytą [9], regulatory PID [10], sterowanie z wyprzedzeniem (feed-forward) [11].

Ostatnie osiągnięcia naukowców [12, 13] pozwalają stwierdzić, że budowa robotów o zmiennej konfiguracji mechanicznej (będąca jak dotąd domeną fantastyki naukowej) staje się rzeczywistością. Celem niniejszej pracy jest przeprowadzenie studium wykonalności robota, na bazie doświadczeń z prac [14, 15], mogącego poruszać się w dwóch trybach jazdy: stabilnym statycznie (poziomym) oraz stabilnym dynamicznie (pionowym), z możliwością automatycznej zmiany trybu ruchu, łącząc zalety obydwu trybów ruchu. System sterowania stanowią algorytmy i regulatory, dzięki którym robot może samodzielnie utrzymywać równowagę w zależności od zadanego sterowania oraz od warunków zewnętrznych, takich jak pochylenie i nierówności podłoża oraz kolizje z przeszkodami.

Artykuł koncentruje się na systemie sterowania, tym niemniej dla kompletności, na wstępie zostaną pokrótce sformułowane wymagania (sekcja 1) oraz konstrukcja mechaniczna i elektroniczna robota (sekcja 2). Postawione wymagania i konstrukcja robota były podstawą do syntezy regulatora kaskadowego (sekcja 3) sterującego silnikami. Struktura regulatora implikowała sposób dekompozycji sterownika sprzętowo-programowego oraz jego implementację na rzeczywistym sprzęcie (sekcja 4). Pracę kończy podsumowanie (sekcja 5).

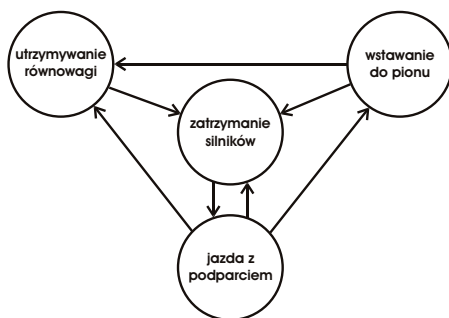
1. Wymagania

Celem projektu było stworzenie taniego robota zdolnego do poruszania się w dwóch trybach ruchu:

- poziomym, czyli stabilnym statycznie, w którym robot ma trzy punkty kontaktu z podłożem: dwa koła oraz zderzak. W tym trybie możliwe jest pchanie przeszkód za pomocą zderzaka, jak również jazda w trudnym terenie,
- pionowym, czyli stabilnym dynamicznie, w którym robot utrzymuje równowagę w pozycji „zderzakiem do góry”. Z założenia ten tryb ma zapewniać zdolność do poruszania się w wąskich przejazdach dzięki małej powierzchni przekroju poziomego, niskie zużycie energii z powodu braku tarcia zderzaka o powierzchnię ziemi, a także wysoką jakość samolokalizacji na bazie odczytów z odometrii.

W obydwu trybach zakładamy zadawanie prędkości postępowej i obrotowej całego robota, a także możliwość automa-

tycznego wstawiania oraz zatrzymania silników w dowolnym trybie ruchu, co ilustruje diagram stanów na rys. 1.



Rys. 1. Diagram stanów, w których może znajdować się robot oraz przejść między stanami

Fig. 1. Robot's state diagram

2. Konstrukcja bazy jezdnej

Konstrukcja mechaniczna (rys. 2) [16] robota nawiązuje do klasy rzucanych robotów zwiadowczych (np. [17]) i składa się z dwóch współosiowych kół zamontowanych na długiej osi, połączonej z ramą. Rozstaw kół jest zbliżony do ich średnicy i wynosi 50 cm. Przeniesienie napędu zostało podobnie rozwiązane jak w rowerze, koła mogą obracać się swobodnie na nieruchomej osi, a napęd jest przenoszony za pomocą zębatek i łańcucha. Do tylnej części ramy przyczepiony jest miękki, owalny zderzak o wysokości zbliżonej do średnicy kół. Element ten stanowi trzeci punkt podparcia, dzięki któremu robot może poruszać się w trybie stabilnym statycznie. Zderzak jest jedynym elementem konstrukcji, który wystaje poza obrys kół. Środek ciężkości robota znajduje się w niewielkiej odległości od osi kół, dzięki czemu robot może samodzielnie wstawać do pozycji pionowej („zderzakiem do góry”) i utrzymywać równowagę poprzez odpowiednie sterowanie. Konstrukcja mechaniczna jest symetryczna w dwóch płaszczyznach:

- prostopadłej do osi kół (symetria lewo – prawo),
- zawierającej oś kół oraz geometryczny środek zderzaka (symetria góra – dół).



Rys. 2. Baza jezdna robota [16]

Fig. 2. Robot's mobile base [16]

Napęd robota stanowią dwa silniki szczotkowe prądu stałego z wkrętarek akumulatorowych, wraz z oryginalnymi

przekładniami planetarnymi i sprzęgłami. Sterowane są zgodnie z metodologią opisaną w pracach [18, 19] poprzez mostek H z dolnymi tranzystorami N-MOSFET i górnymi tranzystorami P-MOSFET. Średni prąd płynący przez każdy z silników mierzony jest pośrednio, poprzez pomiar spadku napięcia na rezystorze o niewielkiej rezystancji. Mostek H stanowi część sterownika silnika, który dodatkowo jest odpowiedzialny za pomiar prędkości obrotowej silnika na podstawie danych z enkoderów, oraz za komunikację z modułem głównym robota. W skład systemu sterowania wchodzi dwa sterowniki silnika.

Za określanie orientacji robota w przestrzeni odpowiada jednostka inercyjna, zawierająca 3-osiowy żyroskop oraz 3-osiowy akcelerometr. Jednostka inercyjna, podobnie jak sterownik silnika, komunikuje się z modułem głównym robota.

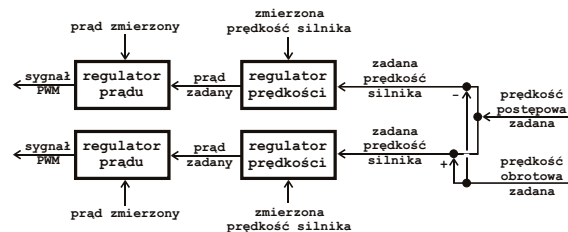
Moduł główny to niewielki netbook o stosunkowo dużej mocy obliczeniowej. Komunikuje się ze wszystkimi podzespołami robota za pomocą magistrali RS-485 oraz RS-232.

3. Struktura regulatora

Do obsługi trybów ruchu z rys. 1 zdecydowano się wykorzystać regulator kaskadowy. Struktura regulatora sterującego ruchem robota jest zależna od aktualnego stanu, w którym znajduje się robot. Jako pętlę najbliższą sprzętowi i zarazem najszybszą przyjęto regulator prądowy, który w naturalny sposób zapewnia ograniczenie prądu płynącego przez silnik, jak również pozwala na pośrednie zadawanie momentu na silniki DC, a więc w pewnym przybliżeniu regulację przyspieszenia robota. Jest to regulator całkujący, którego wejście stanowi uchyb prądu zmierzonego względem zadanego, a wyjściem jest wypełnienie sygnału PWM. Regulator ten jest wspólną częścią wszystkich regulatorów kaskadowych opisanych dalej. Z drugiej strony w najbardziej zewnętrznej pętli zadawana jest prędkość postępową i obrotowa robota.

3.1. Tryb poziomy

W poziomym trybie ruchu zadana prędkość postępową i obrotowa robota jest przeliczana, za pomocą prostych przekształceń, na prędkości obrotowe silników, które z kolei są wejściem dwóch regulatorów kaskadowych, prędkościowo-prądowych (po jednej kaskadzie na każdy silnik – rys. 3). Regulator kaskadowy w trybie poziomym składa się z regulatora całkującego prądowego w wewnętrznej pętli oraz z regulatora proporcjonalnego prędkości obrotowej w zewnętrznej pętli.



Rys. 3. Kaskada regulatorów dla trybu poziomego

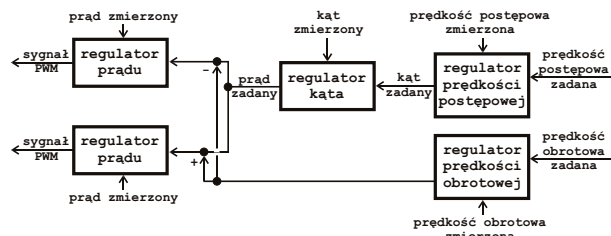
Fig. 3. Cascade of regulators used in horizontal mode

Regulator prędkości pozwala na precyzyjną kontrolę trajektorii robota oraz eliminuje wpływ niewielkiej, trudnej

do uniknięcia asymetrii robota, związanej z różnicami we właściwościach fizycznych silników oraz z różnymi siłami oporu przekładni.

3.2. Tryb pionowy

W trybie pionowym, regulator kaskadowy ma bardziej złożoną strukturę (rys. 4). Wewnętrzne pętle dla każdego z silników stanowią regulatory prądu. Nad nimi działają niezależnie od siebie dwa regulatory: regulator prędkości obrotowej oraz regulator kaskadowy prędkości postępowej i kąta pochylenia.



Rys. 4. Kaskada regulatorów dla trybu pionowego
Fig. 4. Cascade of regulators used in dynamically stable mode

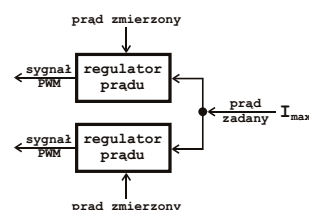
Kaskada regulatorów kąta pochylenia i prędkości postępowej odpowiada za utrzymywanie równowagi oraz za utrzymywanie i zmianę pozycji robota w przestrzeni. Regulator prędkości postępowej to regulator proporcjonalno-całkująco-różniczkujący (PID). Jego wejściem jest uchyb prędkości postępowej zmierzonej względem zadanej. Prędkość postępowa robota została określona jako średnia prędkość obrotowa obu kół. Wyjściem tego regulatora jest zadany kąt dla regulatora kąta pochylenia. Kąt ten jest, dla kątów bliskich kątowi równowagi, wprost proporcjonalny do przyspieszenia robota. Oznacza to, że regulator prędkości postępowej, zadając kąt pochylenia dla regulatora kąta, zadaje przyspieszenie całego robota. Człon różniczkujący pozwala na szybką odpowiedź na zmianę prędkości zadanej, a dzięki członowi całkującemu, robot jest w stanie wjechać na równię pochyłą, nachylając się w stronę równi.

Regulator kąta to regulator proporcjonalno-całkujący. Jego zadaniem jest utrzymywanie zadanego kąta odchylenia robota od pionu, wyznaczonego przez regulator prędkości postępowej. Człon całkujący niweluje uchyb statyczny, który był obserwowany podczas doświadczeń i który był związany z działaniem siły grawitacji na konstrukcję robota. Wyjściem regulatora jest średni prąd zadany dla obu silników.

Niezależnie od regulatora kaskadowego prędkości postępowej i kąta pochylenia, działa regulator proporcjonalno-całkujący prędkości obrotowej robota. Dzięki niemu robot może skręcać oraz utrzymywać zadany kierunek ruchu. Wejściem tego regulatora jest uchyb prędkości obrotowej robota, a wyjściem jest różnicowa wartość prądu dodana do prądu zadanego dla jednego silnika i odjęta od zadanego prądu dla drugiego silnika.

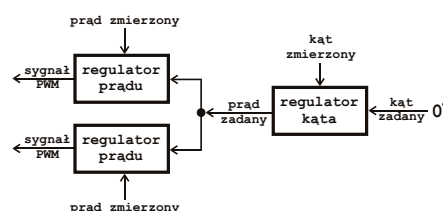
3.3. Wstawanie do pionu

Podczas wstawania do pionu robot wykonuje określoną sekwencję ruchów. W pierwszej fazie manewru robot rozpędza się. Silniki są sterowane prądowo, prądem o maksymalnym dozwolonym natężeniu. Dla tej fazy manewru regulator przedstawiono na rys. 5.



Rys. 5. Regulator dla pierwszej fazy manewru wstawania do pionu
Fig. 5. Regulator for the first phase of getting-up maneuver

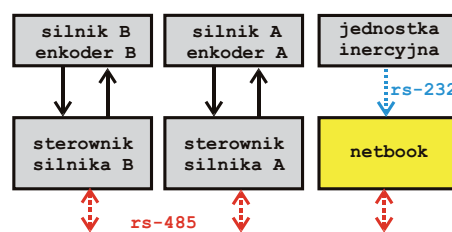
Po upływie określonego czasu, rozpoczyna się druga faza manewru. Polaryzacja prądu zmienia się na przeciwną i zostaje włączony prosty regulator proporcjonalny, który zadaje prąd proporcjonalny do kąta odchylenia od pionu (rys. 6). Druga faza trwa dopóki kąt odchylenia od pionu nie znajdzie się w zdefiniowanym przedziale – robot przechodzi wtedy do stanu balansowania (trybu pionowego).



Rys. 6. Regulator dla drugiej fazy manewru wstawania do pionu
Fig. 6. Regulator for the second phase of getting-up maneuver

4. Dekompozycja i implementacja sterownika

Sterownik programowy robota tworzy system rozproszony (rys. 7), który podzielony został na oprogramowanie niskopoziomowe (sekcja 4.1), działające na sterownikach silników i na jednostce inercyjnej, oraz oprogramowanie wysokopoziomowe (sekcja 4.2) działające na netbooku.



Rys. 7. Moduły systemu
Fig. 7. Modules of the system

Na sterownikach silników zaimplementowano pracujące z dużą częstotliwością regulatory prądu i prędkości obrotowej silników. Na netbooku działa maszyna stanów, która odpowiada za sterowanie całym robotem, oraz regulatory wyższego poziomu: kąta pochylenia, prędkości postępowej i prędkości obrotowej robota.

4.1. Oprogramowanie niskopoziomowe

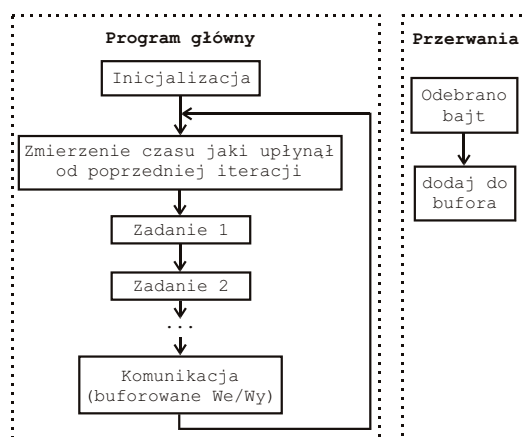
W sterowniku niskopoziomowych zaimplementowano zadania wymagające częstego odwoływania się do sprzętu, z drugiej strony nie wymagające skomplikowanych obliczeń: – komunikację z komputerem PC,

- pomiar czasu,
- sterowanie mostkiem H zasilającym silnik,
- pomiar prądu,
- regulację i ograniczenie prądu,
- pomiar prędkości obrotowej silnika oraz jej regulacja,
- watchdog.

Sterowniki silnika zostały zaimplementowane na mikrokontrolerze AVR ATmega8, który został wybrany ze względu na niską cenę, prostą obsługę i wystarczająco dobre parametry.

Moc obliczeniowa procesora zdeterminowała napisanie aplikacji odwołującej się bezpośrednio do sprzętu bez wykorzystania systemu operacyjnego. Aplikacja została zaprojektowana w taki sposób, aby zadania, wykonywane na jednorzeniowym procesorze, nie blokowały się nawzajem. Ze względu na niewielką liczbę zadań oraz niedużą ich złożoność, możliwe było zastosowanie podejścia zapewniającego względnie częste wykonywanie każdego z zadań. Główna pętla wykonywana jest z dużą częstotliwością, przydzielając każdemu zadaniu czas w każdej pętli. Oprogramowanie (rys. 8) zostało napisane zgodnie z następującymi założeniami:

- żadne zadanie nie stosuje aktywnego oczekiwania i jeśli nie ma w danej chwili niczego do zrobienia, zostaje wstrzymane,
- zadania wykonują jednorazowo niewielką liczbę czynności. W praktyce oznacza to brak pętli o dużej liczbie iteracji,
- stan zadania może zostać zapamiętany na czas wykonywania pozostałych zadań.



Rys. 8. Struktura sterownika silnika
Fig. 8. Diagram of motor controller

Dzięki takiej strukturze aplikacji każde zadanie może być wykonywane z dużą częstotliwością i możliwe jest:

- szybkie pobieranie danych z bufora, aby nie przepełnił się,
- regularne i precyzyjne uruchamianie pewnych procedur, które powinny wykonywać się co stały interwał czasu, np. regulatora prędkości,
- zapamiętanie wyniku konwersji analogowo – cyfrowej tuż po jej zakończeniu i natychmiastowe ponowne uruchomienie kolejnej konwersji,

- ograniczenie liczby przerwań do niezbędnego minimum.

Dodatkowo, zadania mogą działać od siebie niezależnie, co znacznie uprościło aplikację, zmniejszając ryzyko wystąpienia błędów. Nie jest potrzebna synchronizacja między zadaniami, gdyż nie wykonują się one równolegle. Przerwania w większości zastąpiono sprawdzaniem, w każdej iteracji pętli głównej, czy już wystąpiło dane zdarzenie (np. upływanie określonego czasu, lub zakończenie konwersji analogowo-cyfrowej). Poniżej zostaną opisane szczegółowo najistotniejsze zadania sterownika niskopoziomowego.

4.1.1. Regulacja i ograniczenie prądu

Regulacja prądu oparta jest na dyskretnym regulatorze PID, którego wejściem jest uchyb prądu zmierzonego względem zadanego, a wyjściem jest wartość wypełnienia sygnału PWM. Kolejne iteracje regulatora uruchamiane są, z częstotliwością 7,2 kHz, w procedurze obsługi przerwania zegara generującego sygnał PWM. Dzięki temu, w każdej iteracji, wartość na wyjściu regulatora zostaje użyta do sterowania silnikiem. Zastosowano także ograniczenie całkowania, gdy wartość na wyjściu jest zbyt duża. Dodatkowo, można w każdej chwili zmienić wartość jego nastawów, wysyłając odpowiednie dane z komputera PC do sterownika silników.

Regulator prądowy oprócz swojej głównej funkcji, jaką jest sterowanie prądem płynącym przez silnik, zabezpiecza przed spalaniem mostka. W przypadku, gdy nastąpi zwarcie przez np. niepoprawne podłączenie silnika, regulator ograniczy prąd zwarciový i znacznie zmniejszy wzrost temperatury tranzystorów.

4.1.2. Pomiar prędkości obrotowej silnika oraz jej regulacja

Dla trybu jazdy w poziomie regulatory prędkości kół zostały umieszczone w sterowniku niskopoziomowym, w przeciwieństwie do trybu jazdy w pionie, w którym regulatory prędkości umieszczone zostały w sterowniku wysokopoziomowym. Różnica ta wynika ze struktury regulatorów kaskadowych w obu trybach jazdy. W trybie pionowym regulator prędkości działa nad regulatorem kąta, pracującym na netbooku. Dodatkowo, regulator prędkości postępowej steruje średnią prędkością kół robota, dlatego został umieszczony w jednostce centralnej, sterującej całym robotem. W trybie poziomym regulatory prędkości sterują prędkościami obrotowymi kół (poprzez regulację prędkości silników) i mogą działać niezależnie od siebie. Dlatego są implementowane w sterownikach niskopoziomowych, co dodatkowo pozwala na uniknięcie niewielkiego opóźnienia w sterowaniu wynikającego z transmisji danych: prędkości zmierzonej i prądu zadanego.

Regulator PID prędkości obrotowej silnika został zaimplementowany, podobnie jak regulator prądu. Posiada ograniczenie całkowania oraz możliwość zmiany nastaw przez komputer PC. Dodatkowo może zostać w każdej chwili wyłączony, a sterownik silnika przechodzi wtedy w tryb sterowania prądem.

Regulator prędkości działa z częstotliwością około 55 Hz. Tak niska częstotliwość jest skutkiem kompromisu, jaki należało przyjąć wybierając częstotliwość regulacji i dokładność pomiaru. Enkodery generują 1056 impulsów na pełny obrót koła, co przy maksymalnej prędkości robota daje maksymalnie ok. 2000 impulsów na sekundę. Przy 50 pomiarach prędkości na sekundę i maksymalnej prędkości

robotą, zmierzonych zostanie maksymalnie około 40 impulsów. Skutkiem tego jest dość dobra regulacja prędkości robota w zakresie dużych prędkości, zaś nieregularna praca i „szarpanie” przy prędkości bliskiej zeru.

Moment, w którym zostaje uruchomiona kolejna iteracja regulatora prędkości jest określany dzięki zadaniu pomiaru czasu. W kolejnych iteracjach zliczana jest liczba impulsów enkodera (z uwzględnieniem kierunku obrotu) i na tej podstawie obliczana jest prędkość obrotowa silnika. Wejściem regulatora prędkości jest uchyb prędkości zmierzonej względem zadanej, a wyjściem jest wartość prądu, która podana zostaje na wejście regulatora prądu. W trybie jazdy z podparciem (bez utrzymywania równowagi) oba regulatory są włączone tworząc kaskadę.

4.1.3. Watchdog

Współczesne roboty powinny być konstruowane zgodnie z wymogami bezpieczeństwa [20]. Na poziomie mikrokontrolerów mechanizm „watchdog” pozwala na uniknięcie sytuacji, w której mikrokontroler wykonuje bez końca program w pętli (w sposób nieprzewidziany), tj. zawiesi się.

Watchdog zastosowany w programie obsługującym mostek H powoduje reset mikrokontrolera po 0,26 s od czasu ostatniego wyzerowania wewnętrznego timera. Zerowanie timera watchdoga następuje tylko po otrzymaniu poprawnej ramki danych adresowanej do danego urządzenia. Oznacza to, że reset wywołany przez watchdog nastąpi po zawieszeniu się programu, jak również przy przerwie w transmisji danych dłuższej niż 0,26 s.

Zabezpieczenie to zostało przetestowane, kiedy w programie pojawił się błąd, przez który co pewien czas jedno z urządzeń nie przełączało swojego końca magistrali RS-485 w tryb odczytu. Skutkiem tego było zatrzymanie transmisji danych i reset urządzeń wywołany przez watchdoga. Pomimo tego, że zablokowanie magistrali następowało średnio co kilka sekund, możliwe było działanie robota bez większych utrudnień.

4.2. Oprogramowanie wysokopoziomowe

Jednostka centralna to urządzenie, na którym dokonuje się większość obliczeń, gdyż ma największe zasoby ze wszystkich urządzeń w robocie. Jest to netbook z procesorem Intel Atom, 1 GB pamięci RAM oraz dyskiem twardym SSD o pojemności 16 GB. Działa na nim system operacyjny Ubuntu oraz system ROS, stosowane dotychczas z powodzeniem m.in. w systemach nawigacji robotów mobilnych [21].

Gdy robot jest uruchomiony, na jednostce centralnej działa ROS i jego dwa moduły. Jeden moduł obsługuje zadajnik i przesyła dane dotyczące ruchu zadanego przez operatora do drugiego modułu, który steruje całym robotem, realizując dwie najważniejsze funkcje:

- zbieranie i filtrację danych z jednostki inercyjnej,
- zależne od trybu ruchu sterowanie robota na podstawie poleceń od operatora.

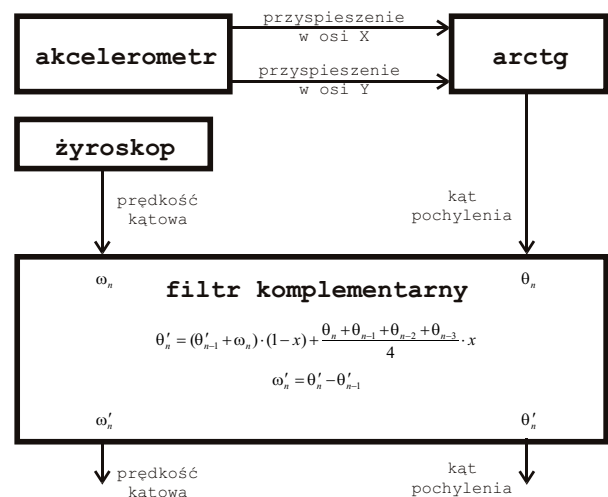
4.2.1. Zbieranie i filtracja danych z jednostki inercyjnej

Moduł główny zawiera regulatory z zamkniętą pętlą sprzężenia zwrotnego i dlatego wymaga ciągłego dostarczania aktualnych danych o stanie fizycznym całego systemu. Wielkości fizyczne są mierzone i zamieniane na postać cyfrową

przez urządzenia wchodzące w skład systemu rozproszonego. Następnie są przesyłane do modułu głównego za pomocą interfejsów komunikacyjnych i tam są ostatecznie przetwarzane.

Sygnały analogowe zamienione na postać cyfrową wymagają dodatkowego przetwarzania przed zastosowaniem ich w systemach regulacji, gdyż zawierają one szумы i błędy pomiarowe, które trzeba wyeliminować lub zmniejszyć. Dodatkowo, w przypadku pomiaru kąta, konieczne jest zastosowanie filtru komplementarnego, łączącego pomiary z co najmniej dwóch czujników.

Na rys. 9 przedstawiono schemat algorytmu filtru komplementarnego, który został zaimplementowany w systemie. Filtr ten zawiera dwa wejścia: ω_n i θ_n oraz dwa wyjścia ω'_n i θ'_n . Zmienne ω i θ to prędkość obrotowa i kąt pochylenia. W wyniku działania filtru otrzymujemy dwa sygnały wyjściowe o podobnej interpretacji co sygnały wejściowe, lecz o znacznie większej dokładności.



Rys. 9. Filtr komplementarny, x to liczba z przedziału (0; 1)
Fig. 9. Complementary filter, x is a real number in (0; 1)

Filtr ten jest dyskretny w dziedzinie czasu, tzn. wykonywany jest w iteracjach w stałych przedziałach czasu. Jest podany wzorem rekurencyjnym na rys. 9. Wewnętrzny stan filtru jest przechowywany, między kolejnymi iteracjami, za pomocą zmiennych θ'_{n-1} , θ_{n-1} , θ_{n-2} oraz θ_{n-3} . Liczba x to stała z przedziału (0; 1), która określa wagę wejść. Wartość stałej x bliska 0 oznacza, że wartości na wyjściach będą wyliczane głównie na podstawie całki prędkości kątowej. Wartość x bliska 1 oznacza, że wartości na wyjściu będą bardziej wrażliwe na pomiar kąta z akcelerometru. Im mniejszy dryf żyroskopu, tym mniejsza może być wartość stałej x i tym większa dokładność pomiaru kąta. W przedstawionej tutaj implementacji dobrano wartość $x = 0.03$, przy której dryf żyroskopu nie wprowadza błędu.

Otrzymana na wyjściu wartość kąta pochylenia jest znacznie dokładniejsza i bliższa rzeczywistej wartości kąta. Wyniki badania działania filtru komplementarnego zostaną przedstawione w kolejnej części artykułu. Uzyskana wartość kąta pochylenia robota jest wykorzystywana w pętli sprzężenia zwrotnego regulatora kąta.

4.2.2. Poziomy tryb ruchu

W trybie jazdy w poziomie, moduł główny przelicza dane z zadajnika na prędkości obrotowe kół, które następnie przesyła do kaskad regulatorów w sterownikach silników. W trybie tym, w module głównym nie działa żaden regulator.

4.2.3. Pionowy tryb ruchu

W trybie jazdy w pionie moduł główny zawiera część kaskady regulatorów: regulator kąta odchylenia od pionu oraz regulatory prędkości postępowej i obrotowej robota.

Regulator kąta jest dyskretnym regulatorem PI, a regulator prędkości postępowej jest dyskretnym regulatorem PID. Oba regulatory tworzą kaskadę i działają z częstotliwością 100 Hz. Zaimplementowane ograniczenie całkowania oraz ograniczenie wartości na wyjściu zabezpieczają system przed zbyt gwałtowną i zbyt silną odpowiedzią na sytuacje wyjątkowe, jak np. nagła utrata stabilności.

Regulatory te są uruchomione tylko w trybie utrzymywania równowagi w pozycji pionowej i są resetowane przy każdym przejściu do tego trybu. Reset polega na wyzerowaniu całki, aby w początkowych iteracjach regulacji nie była brana pod uwagę jej wartość z poprzedniej fazy jazdy w trybie utrzymywania równowagi.

Oprócz sterowania prędkością postępową robota konieczne jest także sterowanie prędkością obrotową robota w trybie balansowania. W tego typu konstrukcjach stosuje się napęd różnicowy, w którym sterowanie prędkością obrotową robota polega na zadawaniu różnych prędkości dla każdego z kół. W przypadku opisanego w niniejszym artykule robota, będącego w trybie jazdy w pozycji pionowej, nie można bezpośrednio i precyzyjnie kontrolować różnicy w prędkości obrotowej kół, gdyż regulator prędkości postępowej jest wspólny dla obu kół. Z kolei regulator kąta zadaje prąd płynący przez oba silniki, co przy nawet niewielkiej asymetrii w elementach mechanicznych lub elektronicznych skutkuje w nieznacznych różnicach w prędkości obrotowej silników i niekontrolowanym skręcaniu robota.

Aby poprawić kontrolę nad trajektorią robota, dodano jeszcze jeden regulator PI, który steruje jego prędkością obrotową podczas ruchu w pozycji pionowej. Jego wyjście stanowi różnicowa wartość prądu, która jest dodawana do wartości prądu wyznaczonej przez regulator kąta dla jednego silnika i odejmowana od tejże wartości dla drugiego silnika.

4.2.4. Manewr wstawania do pionu

Poszczególne fazy manewru są zlecane ze sterownika wysokopoziomowego. Pierwsza, w której robot się rozpędza w pozycji poziomej trwa 0,7 s. Silniki są sterowane prądowo, prądem o maksymalnym dozwolonym natężeniu. Następnie, po czasie 0,7 s, rozpoczyna się druga faza, gdzie polaryzacja prądu zmienia się na przeciwną i zostaje włączony prosty regulator proporcjonalny, który zadaje prąd proporcjonalnie do kąta odchylenia od pionu. Również w tej fazie zadane natężenie prądu podlega ograniczeniu. Druga faza trwa dopóki kąt odchylenia od pionu nie znajdzie się w przedziale ustalonym eksperymentalnie ($-10^\circ \dots +10^\circ$). Jeżeli po czasie 2,0 s od rozpoczęcia manewru robot nie osiągnie trybu pionowego (czyli nie osiągnie założonego kąta), zostaje włączony stan awaryjnego zatrzymania sil-

ników. Czasy trwania obu faz manewru zostały dobrane doświadczalnie. Zachowanie robota podczas manewru wstawania do pionu zostanie szczegółowo opisane w kolejnej części artykułu.

5. Podsumowanie

Zaprojektowany i wykonany w ramach projektu robot może poruszać się w dwóch trybach jazdy, automatycznie zmieniać tryb ruchu, wjeżdżać na równię pochyłą, pchać obiekty, utrzymywać równowagę po zderzeniu z przeszkodą oraz zachowuje się w sposób przewidywalny i bezpieczny. Należy zwrócić szczególną uwagę na niezawodność robota, który podczas testów nie ulegał awariom oraz nie stanowił zagrożenia dla otoczenia. Jego system sterowania został gruntownie przetestowany w różnych scenariuszach. W czasie testów zbierano i analizowano dane szczegółowo opisujące stan robota. Przeprowadzone doświadczenia, wraz z przebiegami czasowymi zmiennych stanu robota, zostaną przedstawione w kolejnym artykule. Z kolei badania filtracji danych z jednostki inercyjnej będą podstawą do rozwinięcia metod estymacji siły bezwładności [22].

Podziękowania

Tomasz Winiarski dziękuje za wsparcie otrzymane w ramach stypendium współfinansowanego przez Unię Europejską w ramach Europejskiego Funduszu Społecznego, które przyznawane jest przez Centrum Studiów Zaawansowanych Politechniki Warszawskiej w ramach projektu „Program Rozwojowy Politechniki Warszawskiej”. Projekt był finansowany ze środków Narodowego Centrum Nauki przyznanych na podstawie decyzji numer DEC-2012/05/D/ST6/03097.

Bibliografia

1. Klimasara W., *Koncepcja, projekt oraz konstrukcja mechaniczna mobilnego robota interwencyjno-inspekcyjnego sr-10 inspector*, "Krajowa Konferencja Robotyki, VII KKR, Prace Naukowe ICT PWr. Proceedings Red. K. Tchon, Ładek Źródł", 2001, 139–148.
2. Nguyen H. G., Morrell J., Mullens K., Burmeister A., Miles S., Farrington N., Thomas K., Cage D. W. (2004): *Segway robotic mobility platform*. Defense Technical Information Center.
3. Kędzierski J., Małek L., Oleksy A., *Budowa robota społecznego FLASH*, [in:] XII Krajowa Konferencja Robotyki – Postępy Robotyki, Oficyna Wydawnicza Politechniki Warszawskiej, 2012, 681–694.
4. Stilman M., Olson J., Gloss W. (May, 2010): *Golem Krang: Dynamically Stable Humanoid Robot for Mobile* William Gloss, [in:] IEEE International Conference on Robotics and Automation ICRA'10, IEEE.
5. Cazzolato B., Harvey J., Dyer C., Fulton K., Schumann E., Zhu T., Prime Z., *Control of an electric diwheel*, 2009.
6. McComb G. (2001): *The robot builder's bonanza*. McGraw-Hill/TAB Electronics.
7. Ooi R. C., *Balancing a two-wheeled autonomous robot*, "University of Western Australia", 2003.
8. Kim Y., Kim S. H., Kwak Y. K., *Dynamic analysis of a nonholonomic two-wheeled inverted pendulum robot*, "Journal of Intelligent & Robotic Systems" 1/2005, 25–46.

9. Li Z., Xu C., *Adaptive fuzzy logic control of dynamic balance and motion for wheeled inverted pendulums*, "Fuzzy Sets and Systems" 12/2009, 1787–1803.
10. Chang W.-D., Hwang R.-C., Hsieh J.-G., *A self-tuning PID control for a class of nonlinear systems based on the Lyapunov approach*, "Journal of Process Control" 2/2002, 233–242.
11. Mazenc F., Bowong S., *Tracking trajectories of the cart-pendulum system*, "Automatica" 4/2003, 677–684.
12. Hajkuś J., Machnik M., Zając J., *IRMO-bot inspekcyjno-ratunkowy, rekonfigurowalny robot mobilny*, [in:] XII Krajowa Konferencja Robotyki – Postępy Robotyki, Oficyna Wydawnicza Politechniki Warszawskiej, 2012, 513–524.
13. Trojnecki M., Skrzypczyński P., *Projekt konstrukcji robota mobilnego o zmiennej konfiguracji*, [in:] XII Krajowa Konferencja Robotyki – Postępy Robotyki, Oficyna Wydawnicza Politechniki Warszawskiej, 2012, 549–557.
14. Zieliński C., Kornuta T., Trojanek P., Winiarski T., *Metoda projektowania układów sterowania autonomicznych robotów mobilnych. Część 1. Wprowadzenie teoretyczne*, "Pomiary – Automatyka – Robotyka PAR" 9/2011, 84–87.
15. Zieliński C., Kornuta T., Trojanek P., Winiarski T., *Metoda projektowania układów sterowania autonomicznych robotów mobilnych. Część 2. Przykład zastosowania*, "Pomiary – Automatyka – Robotyka PAR" 10/2011, 84–91.
16. Seredyński D., Winiarski T., Banachowicz K., Wałęcki M., Stefańczyk M., Majcher P., *Robot mobilny o zmiennym sposobie lokomocji – konstrukcja mechaniczna i elektroniczna*, "Pomiary – Automatyka – Robotyka PAR" 1/2013, 162–167.
17. Czupryniak R., Trojnecki M., *Throwable tactical robot – description of construction and performed tests*, "Journal of Automation Mobile Robotics and Intelligent Systems" 4/2010, 26–32.
18. Wałęcki M., Banachowicz K., Winiarski T., *Research oriented motor controllers for robotic applications*, [in:] Kozłowski K. (ed.), Robot Motion and Control 2011 (LNCIS) Lecture Notes in Control & Information Sciences, Springer Verlag London Limited, 2012, 193–203.
19. Wałęcki M., Banachowicz K., Stefańczyk M., Chojęcki R., Wiśniowski M., Winiarski T., *Uniwersalna struktura sprzętu badawczo-dydaktycznej platformy mobilnej*, [in:] XII Krajowa Konferencja Robotyki – Postępy Robotyki, Oficyna Wydawnicza Politechniki Warszawskiej, 2012, 305–314.
20. Missala T., *Safety of robots in a neighborhood of the people and the new law of robotics*, "Pomiary, Automatyka, Robotyka" 1/2012, 48–52.
21. Stefańczyk M., Banachowicz K., Wałęcki M., Winiarski T., *Nawigacja robotem Elektron z wykorzystaniem kamery 3D i lidar*, [in:] XII Krajowa Konferencja Robotyki – Postępy Robotyki, Oficyna Wydawnicza Politechniki Warszawskiej, 2012, 265–274.
22. Winiarski T., Banachowicz K., *System akwizycji skorygowanej siły uogólnionej kontaktu robota manipulacyjnego z otoczeniem*, "Pomiary – Automatyka – Robotyka PAR" 2/2013, 390–394.

Zobacz więcej

Pobierz bezpłatną aplikację PAR+
App Store | Google Play



Mobile robot with two modes of locomotion – control system

Abstract: The article describes control system of mobile robot with two coaxial wheels, that can move in two modes of locomotion: dynamically stable and statically stable. The robot can switch between these two modes with automatic maneuver. The control system is based on cascades of PID regulators.

Keywords: mobile robot, inverted pendulum, PID cascade

mgr inż. Dawid Seredyński

W 2012 r. ukończył Wydział Elektroniki i Technik Informacyjnych Politechniki Warszawskiej. Był aktywnym członkiem koła robotyki „Bionik” na macierzystym wydziale, gdzie tworzył i rozwijał układy sterowania robotów mobilnych. Podsumowaniem badań prowadzonych w trakcie studiów była praca magisterska „System sterowania dwukołowym robotem mobilnym o zmiennym sposobie lokomocji”, która została obroniona z wyróżnieniem, a rok później uzyskała także wyróżniona w V edycji konkursu Młodzi Innowacyjni PIAP.
e-mail: dawid.seredynski@gmail.com



dr inż. Tomasz Winiarski

Jest adiunktem w Instytucie Automatyki i Informatyki Stosowanej Politechniki Warszawskiej. Sprawuje funkcje kierownika laboratorium robotyki w macierzystym instytucie, a także opiekuna studenckiego koła naukowego Bionik, które współtworzył i z którym organizował imprezy popularizujące robotykę oraz realizował granty badawcze. W 2010 r. otrzymał za osiągnięcia naukowe nagrodę indywidualną drugiego stopnia rektora PW, w 2011 r. wyróżnienie w konkursie Innowator Mazowsza, a także pierwszą nagrodę w konkursie Młodzi Innowacyjni PIAP. Jego zainteresowania naukowe dotyczą z jednej strony konstrukcji i nawigacji robotów mobilnych dedykowanych do zadań usługowych, z drugiej strony specyfikacji i implementacji zadań manipulatorów i chwytaków ze szczególnym uwzględnieniem hybrydowego sterowania pozycyjno-siłowego oraz sterowania impedancyjnego.

e-mail: tmwiniarski@gmail.com

