

POLITECHNIKA WROCŁAWSKA
WYDZIAŁ ELEKTRONIKI

KIERUNEK: Automatyka i Robotyka (AIR)
SPECJALNOŚĆ: Robotyka (ARR)

**PRACA DYPLOMOWA
INŻYNIERSKA**

Rola konfiguracji czujników linii w procesie
sterowania robota klasy linefollower

Line sensors configuration
for linefollower robots control

AUTOR:
Michał Orczyk

PROWADZĄCY PRACĘ:
dr inż. Robert Muszyński, K-7

OCENA PRACY:

Spis treści

1	Wstęp	3
2	Konstrukcja robota	5
2.1	Konstrukcja mechaniczna	5
2.2	Układy elektroniki	7
2.3	Wykrywanie położenia linii	7
2.4	Wyznaczanie prędkości obrotowej kół	9
2.5	Regulatory prędkości obrotowej kół	10
2.6	Regulator położenia względem linii	10
2.6.1	Modyfikacja regulatora dla powolnych przejazdów	11
3	Mapowanie trasy	13
3.1	Reprezentacja trasy i położenia robota	13
3.1.1	Reprezentacja w postaci współrzędnych kartezjańskich	14
3.1.2	Reprezentacja w postaci krzywizny	14
3.2	Budowa mapy trasy	15
3.2.1	Analiza błędów mapowania	16
4	Konfiguracja czujników w procesie sterowania	19
4.1	Wpływ dokładności wyznaczania trasy na poprawność optymalizacji przejazdu	19
4.2	Wyznaczanie profilu prędkości w zależności od konfiguracji	20
5	Badania eksperymentalne	23
5.1	Wpływ konfiguracji czujników na dokładność mapowania	23
5.2	Wpływ konfiguracji czujników na dokładność mapowania	25
5.3	Analiza wyników mapowania trasy pod kątem optymalizacji przejazdu	27
5.4	Profil prędkości robota w zależności od konfiguracji czujników	29
6	Podsumowanie	33
	Bibliografia	34

Rozdział 1

Wstęp

Robot klasy linefollower to robot, którego zadaniem jest podążanie za czarną linią na kontrastującym podłożu. Roboty tej klasy są najczęściej budowane w celu udziału w zawodach robotycznych, podczas których zadaniem jest wykonanie przejazdu w najkrótszym możliwym czasie. Roboty podążające za linią są najczęściej klasy (2,0), co oznacza, że robot posiada dwa koła napędowe sztywno zamocowane do korpusu konstrukcji. Sterowanie robotem jest sterowaniem różnicowym, a więc zmiana kierunku poruszania się robota wynika z różnicy prędkości obrotu kół. Takie sterowanie sprawia, że robot może skręcać w miejscu, a co za tym idzie, sprawdza się dobrze na skomplikowanych trasach z dużą liczbą często występujących zakrętów.

Roboty, których celem jest uczestnictwo w zawodach, muszą spełniać ściśle określone wymagania konstrukcyjne zawarte w regulaminie zawodów [2]. Wymagania te ograniczają możliwości konstrukcyjne robotów poprzez kryteria odnośnie wymiarów oraz możliwości komunikacji z robotem. Jednym z wymogów jest maksymalny wymiar robota, którego obrys musi zawierać się w obszarze kartki papieru o rozmiarze A4 — waga i wysokość robota nie są ograniczone. Podczas przejazdów nie wolno komunikować się z robotem za wyjątkiem komunikatów, które dotyczą startu i zatrzymania robota.

W typowych rozwiązaniach roboty klasy linefollower pozyskują informację o położeniu linii z czujników równomiernie rozmieszczonych na listwie. Standardowe konstrukcje robotów tej klasy najczęściej są wyposażone w sterownik PID, kontrolujący położenie względem linii. Tego typu rozwiązanie sprawdza się dobrze, ponieważ robot zawsze dąży do tego, aby utrzymać swój obrys na linii, co jest wymogiem wykonania poprawnego przejazdu podczas zawodów. Ten sposób sterowania nie wykorzystuje jednak w pełni możliwości robota, ponieważ do robota dostarczane są informacje o przebiegu trasy z niewielkim wyprzedzeniem, które wynika z wysunięcia listwy z czujnikami. Ponadto, w takim rozwiązaniu robot podąża dokładnie za linią, nie biorąc pod uwagę faktu, że pewne fragmenty toru można pokonać w prostszy sposób.

W celu zwiększenia możliwości robotów, czyli skrócenia czasu ich przejazdów, stosuje się algorytmy optymalizacji. Jednym z takich algorytmów jest algorytm profilowania prędkości [9], którego zadaniem jest dobór odpowiednich prędkości do danych fragmentów toru. Istnieją też inne strategie optymalizacji, takie jak optymalizacja toru ruchu robota [5], które optymalizują kształt toru ruchu robota. Optymalizacja kształtu ma na celu zmniejszenie krzywizny toru ruchu oraz skrócenie dystansu, który musi pokonać robot, aby w rezultacie skrócić czas przejazdu. Al-

gorytmy optymalizacji wymienione powyżej wymagają jednak apriorycznej wiedzy na temat przebiegu toru. Wiedzę taką pozyskuje się poprzez wykonanie próbnego przejazdu mapującego, podczas którego lokalizowane są kolejne punkty trasy.

Działanie algorytmu optymalizacji zależy w bardzo dużym stopniu od wiedzy o torze, który ma pokonać robot. W rzeczywistych warunkach wyznaczenie przebiegu toru w sposób idealny jest niemożliwe. Poprawę technik mapowania można uzyskać poprzez zastosowanie czujników o większej dokładności, zwiększenie ich liczby lub poprzez wydajniejsze metody przetwarzania informacji pochodzących z czujników. Lepsze wykorzystanie informacji pochodzących z czujników przekłada się na zmniejszenie błędu z jakim mapowana jest trasa. Poprawia to z kolei jakość sterowania robotem na podstawie przebiegu mapy trasy.

Celem pracy jest zbadanie wpływu położenia czujników linii na proces mapowania trasy przejazdu oraz jej realizację. Dla różnych wysunięć czujników błąd określania orientacji robota przekłada się w różnym stopniu na dokładność z jaką wyznaczana jest trasa. Rozpatrując problem mapowania trasy pod kątem optymalizacji, pożądane jest, aby wyznaczana mapa była możliwie najdokładniejsza. Z tego powodu przeprowadza się badania, które mają na celu zbadanie wpływu zmiany konfiguracji czujników na proces sterowania robotem.

Zasadniczym celem pracy jest ocena wpływu konfiguracji czujników na poszczególne etapy optymalizacji, a przede wszystkim udzielenie odpowiedzi na następujące pytania badawcze:

- jak zmienia się wartość błędu mapowania trasy w zależności od konfiguracji czujników,
- jakie korzyści mogą wynikać ze zmiany konfiguracji czujników w procesie sterowania robotem.

Układ pracy jest następujący. Rozdział 2 poświęcono opisowi robota oraz algorytmom sterowania robotem. W rozdziale 3 omówiono proces mapowania oraz analizę błędów mapowania trasy. Rozdział 4 rozpatruje wpływ wysunięcia listwy z czujnikami na proces optymalizacji przejazdu. W rozdziale 5 przedstawiono wyniki przeprowadzonych badań dotyczących wpływu położenia listwy z czujnikami na proces mapowania trasy oraz proces sterowania robotem. W rozdziale 6 podsumowano rezultaty przeprowadzonych badań, przedstawiono napotkane trudności oraz zaproponowano potencjalny obszar dalszych badań.

Rozdział 2

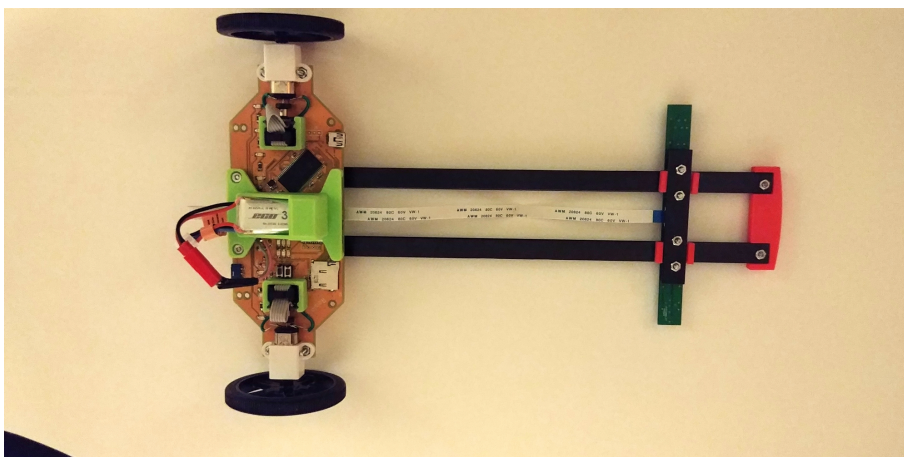
Konstrukcja robota

Prezentowany w niniejszej pracy robot klasy linefollower, jak wiele tego typu konstrukcji, jest robotem klasy (2,0). Jest on sterowany różnicowo, co przekłada się na możliwość skręcania w miejscu. Cechą charakterystyczną robota jest możliwość zmiany położenia listwy z czujnikami bez ingerencji w jego konstrukcję. Robot posiada interfejsy komunikacyjne, czujniki stanu wewnętrznego oraz zewnętrznego. Odczyty z czujników najczęściej wymagają wstępnego przetworzenia do formy, która jest użyteczna podczas dalszego procesu sterowania robotem.

2.1 Konstrukcja mechaniczna

Konstrukcja robota składa się z płyty głównej, przedniej listwy zawierającej czujniki linii, a także z elementów mocujących (zobacz rysunki 2.1, 2.2). Płyta główna, będąca jednocześnie układem elektronicznym i elementem konstrukcyjnym, łączy ze sobą pozostałe elementy robota. Napęd robota stanowią dwa silniki Pololu HPCB z przekładnią 10:1 [13], które zostały symetrycznie rozmieszczone po obu stronach płyty głównej i na osiach których zamontowano bezpośrednio dwa koła o średnicy 60mm. Zasilająca robota bateria została zamocowana w specjalnie zaprojektowanym koszyku, który pozwolił zainstalować ją ponad elementami elektronicznymi robota. Koszyk został zaprojektowany tak, aby środek baterii znajdował się ponad środkiem płyty głównej robota. Usytuowanie baterii w środku robota oraz symetryczność płyty głównej robota mają przelożyć się na zmniejszenie momentu bezwładności robota. Umieszczenie baterii ponad płytą główną robota przekłada się na mniejsze rozmiary całej konstrukcji, poprzez lepsze rozmieszczenie elementów elektronicznych.

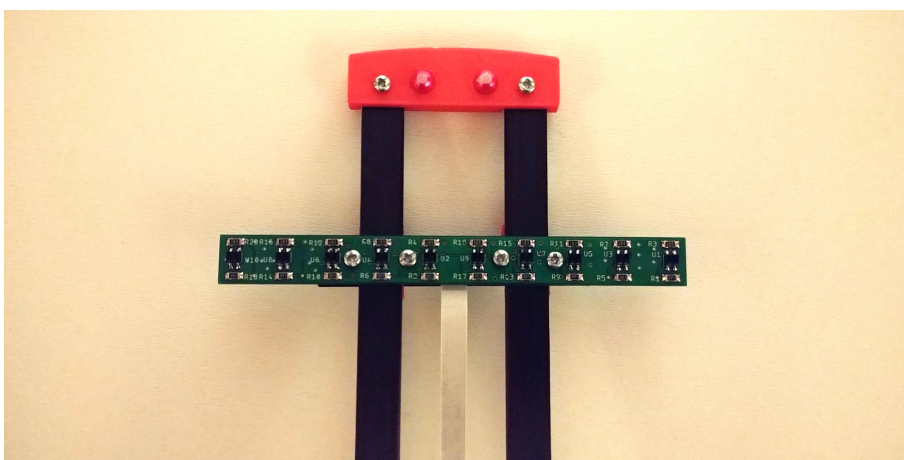
Robot został zaprojektowany tak, aby była możliwa zmiana położenia czujników linii względem jego środka. Ruch przedniej listwy (rysunek 2.3), na której zainstalowane są czujniki, możliwy jest dzięki prowadnicy. Listwa z zainstalowanymi czujnikami zostaje unieruchomiona poprzez dokręcenie czterech śrub do elementu obejmującego. Element obejmujący został wykonany z listwy węglowej oraz został wyposażony w dwa elementy wykonane za pomocą druku 3D, które zapobiegają ślizganiu się. Prowadnica została wykonana z dwóch listw węglowych zapewniających lekkość i wytrzymałość. Prowadnica została zamontowana w tylnej części robota na wysięgniku, aby możliwe było ustawienie listwy z czujnikami również wzdłuż osi poprzecznej robota. Średnicę kół robota (60mm) dobrano tak, aby prowadnica znajdowała się w dostatecznie dużej odległości od płyty głównej robota. Rozmiar kół,



Rysunek 2.1 Widok robota z góry



Rysunek 2.2 Widok robota z boku



Rysunek 2.3 Ruchomy moduł czujników linii

własności czujników linii oraz założenie odnośnie równoległości prowadnicy do podłoża zdeterminowały również wysokość ślizgacza, którym zakończona jest prowadnica. Elementy konstrukcyjne takie jak ślizgacz, elementy antypoślizgowe, objemki, wysięgnik oraz uchwyty enkoderów zostały wykonane przy użyciu druku 3D, metodą przyrostową FDM. Elementy drukowane zostały wykonane z materiału PLA.

2.2 Układy elektroniki

Układ elektroniki robota stanowią cztery płyty PCB, w których skład wchodzi płyta główna robota, płyta zawierająca czujniki linii oraz dwie płyty z zainstalowanymi enkoderami. Płyta główna robota zawiera jednostkę obliczeniową, układy pomiaru stanu wewnętrznego i zewnętrznego robota, a także interfejsy komunikacyjne.

Rolę jednostki sterującej pełni mikrokontroler *stm32f767zit6* [18], który posiada sprzętowe wsparcie operacji zmiennoprzecinkowych pojedynczej i podwójnej precyzji. Mikrokontroler został skonfigurowany tak, że częstotliwość taktowania zegara systemowego wynosi 216MHz . Źródłem taktowania mikrokontrolera jest zewnętrzny rezonator kwarcowy o częstotliwości rezonansowej 16MHz . Całkowita pamięć mikrokontrolera składa się z 2MB pamięci Flash oraz z 512KB pamięci SRAM.

Na środku płyty głównej robota znajduje się żyroskop *L3GD20* [16], którego zakres został skonfigurowany na $\pm 250^\circ/s$, co przekłada się na czułość $0.00875^\circ/s$. Część układu pomiarowego stanowi również układ *LSM303C* [17], który zapewnia trzyosiowy pomiar przyspieszenia oraz pomiar kierunku pola magnetycznego. Zakres pomiarowy akcelerometru został skonfigurowany na $\pm 2g$, co przekłada się na czułość 0.061mg . Zakres pomiaru natężenia pola zawiera się w przedziale $\pm 1.6\text{mT}$, a czułość magnetometru wynosi 58pT .

Pomiar przemieszczenia kół jest dokonywany za pomocą enkoderów magnetycznych *AM4096* [14], których rozdzielczość wynosi 12-bitów na obrót wału silnika. Odpowiada to 40960 impulsów na obrót koła. Wiedząc, że średnica koła wynosi 60mm możemy obliczyć niepewność określania położenia na poziomie $4,7\mu\text{m}$.

Położenie linii jest wykrywane za pomocą listwy sensorycznej, w której skład wchodzi dziesięć czujników uszeregowanych w linii prostej w odstępach 12mm . Czujnikami linii są transoptory odbiciowe *KTIR0711S* [7]. Linia jest wykrywana na podstawie zmiany napięcia, wynikającego z ilości światła odbitego, powracającego do czujnika. Zmiana napięcia jest mierzona przez 12-bitowy przetwornik ADC, a następnie przekazywana do pamięci mikrokontrolera z wykorzystaniem DMA.

Komunikacja bezprzewodowa, umożliwiająca wymianę danych z urządzeniami zewnętrznymi, została zrealizowana z wykorzystaniem modułu bluetooth *RN4678* [11], którego prędkość transmisji skonfigurowano na $115,2\text{kb/s}$. Moduł bluetooth może pracować w trybie niskiego poboru energii lub klasycznym. Na potrzeby pracy moduł bluetooth został skonfigurowany tak, aby pracował w trybie klasycznym. Komunikacja z modułem odbywa się za pomocą protokołu UART, z wykorzystaniem mechanizmu kontroli przepływu *RTS/CTS*.

2.3 Wykrywanie położenia linii

Wykrywanie położenia linii jest procesem przetwarzania odczytów z czujników linii na jej położenie, które jest określane względem środka listwy zawierającej czujniki.

Punkt, w którym znajduje się linia jest określany na podstawie odczytów z n czujników. Wyznaczanie położenia linii składa się z kilku etapów, dzięki którym na podstawie odczytów z czujników odbiciowych jesteśmy w stanie określić położenie linii, a mianowicie z:

- odwrócenia kierunku przyrostu wartości mierzonych,
- normalizacji pomiarów,
- wyznaczania położenia linii względem środka listwy sensorycznej.

Położenie linii wykrywane jest za pomocą transoptorów odbiciowych, które składają się z diody emitującej światło podczerwone oraz fototranzystora. Wykrywanie linii odbywa się poprzez pomiar napięcia na fototranzystorze. Mierzone napięcie jest większe, gdy czujnik znajduje się nad powierzchnią dobrze pochłaniającą światło podczerwone niż w sytuacji, gdy czujnik znajduje się nad powierzchnią odbijającą światło w większym stopniu. Wygodnie jest przekształcić pomiary tak, aby kierunki zmian otrzymanych wartości w_i były przeciwne do zmian kierunków wartości mierzonych. Zmiana kierunku przyrostu odczytywanych pomiarów sprawia, że narastają one od zerowego poziomu. Wartości są dostosowane do dalszego przetwarzania, ułatwiając normalizację. Napięcia odczytywane z czujników przekształcamy wykorzystując równanie

$$w_i = (u_{max} - u_i), \quad i = 1 \dots n, \quad (2.1)$$

gdzie u_i są mierzonymi napięciami i -tego czujnika, a u_{max} maksymalnym poziomem napięcia, jaki jest osiągalny podczas pomiaru.

Ostatecznie, pomiary zostają przekształcane do postaci znormalizowanej p_i . Normalizacja odczytów jest konieczna, aby zakres mierzonych wartości był stały dla wszystkich czujników [6]. Konwersji można dokonać wykorzystując równanie

$$p_i = \lambda_i w_i, \quad (2.2)$$

gdzie λ_i są współczynnikami normalizującymi odczyty z czujników. Współczynniki normalizujące są wykorzystywane do normalizacji odczytów z czujników w trakcie każdego z przejazdów robota. Kalibracja pozwala na dokonywanie poprawnej normalizacji odczytów niezależnie od wysunięcia listwy sensorycznej względem środka robota. Zmiana wysunięcia może powodować niewielkie zmiany odległości między czujnikami linii a podłożem, co może wpływać na przeskalowanie wartości znormalizowanych. Z tego powodu kalibracja jest dokonywana każdorazowo po zmianie wysunięcia poprzez umieszczenie robota na białym podłożu, a następnie wyznaczenie współczynników λ_i według następującej formuły:

$$\lambda_i = \frac{1}{w_{w,i}}, \quad (2.3)$$

gdzie $w_{w,i}$ jest zarejestrowanym napięciem przez i -ty czujnik na białym podłożu.

Znormalizowane pomiary pozwalają na zastosowanie metody wierzchołka paraboli [6], w celu wykrywania położenia linii. W tej metodzie położenie linii jest wyznaczane przez wierzchołek paraboli przechodzącej przez punkt reprezentujący najmniejszą wartość spośród wszystkich wartości odczytanych z czujników oraz przez punkty reprezentujące odczyty z dwóch sąsiednich czujników. Pozycja linii jest wyznaczana z uwzględnieniem stałej odległości między czujnikami przy założeniu, że

czujniki są usytuowane względem siebie w linii prostej. Pozycja punktu określającego środek linii h względem osi symetrii robota jest obliczana na podstawie równania

$$h = d \frac{1}{2} \frac{p_{k-1} - p_{k+1}}{p_{k-1} - 2p_k + p_{k+1}} + d \left(k - \frac{1}{2} (n + 1) \right), \quad p_k = \min_{i=2, \dots, n-1} (p_i), \quad (2.4)$$

gdzie d jest odległością pomiędzy czujnikami wyrażaną w milimetrach.

2.4 Wyznaczanie prędkości obrotowej kół

Prędkości obrotu kół są wyznaczone w sposób pośredni na podstawie ich przemieszczenia kąowego. Przemieszczenia kąowe są określane z ograniczoną dokładnością spowodowaną ograniczoną rozdzielczością enkoderów, co przekłada się na niepewność określania prędkości obrotowych kół. Pośrednie pomiary prędkości są obarczone dużymi błędami spowodowanymi błędami pomiaru czasu oraz przemieszczenia, dlatego istotne jest zastosowanie metody pomiaru pośredniego o większej dokładności pomiaru. Istnieją różne metody pomiaru pośredniego, których prostocie nie zawsze towarzyszy duża dokładność pomiaru [10, 12]. Metoda wykorzystująca iloraz różnicowy do wyliczania kolejnych pochodnych na podstawie chwilowych przemieszczeń kąowych, jest obciążona zbyt dużymi błędami pomiarowymi, które uniemożliwiają wykorzystanie tak obliczonej prędkości w sprzężeniu zwrotnym regulatora prędkości obrotowej kół. Znacznie lepszym rozwiązaniem jest wykorzystanie metody, która opiera się na dopasowaniu prostej stycznej do wykresu przyrostu położenia kąowego koła [10]. Metoda oparta na graficznej interpretacji prędkości jako tangens kąta nachylenia prostej stycznej do wykresu położenia kąowego koła sprawia, że prędkości mogą być wyznaczone na bazie większej liczby pomiarów, a co za tym idzie, prędkości są wyznaczone znacznie dokładniej i są odporne na chwilowe zaburzenia odczytów.

W celu wyznaczenia prędkości obrotowej pojedynczego koła z wykorzystaniem metody dopasowania prostej, należy wyznaczyć współczynnik kierunkowy prostej stycznej do wykresu przemieszczenia kąowego koła. Przebieg przemieszczeń kąowych w czasie nie jest jednak idealny, w rezultacie będziemy wyznaczać prostą najlepiej dopasowaną w danym oknie czasowym. Współczynnik kierunkowy prostej traktujemy jako pochodną przemieszczenia kąowego, czyli prędkość. Współczynnik został wyznaczony przy wykorzystaniu metody najmniejszych kwadratów [8].

Dla wektorów

$$x = [x_1, x_2, \dots, x_m]^T \quad \text{i} \quad y = [y_1, y_2, \dots, y_m]^T, \quad (2.5)$$

definiowany jest iloczyn skalarny

$$\langle x, y \rangle = \sum_{i=1}^m x_i y_i, \quad (2.6)$$

gdzie m jest szerokością okna czasowego wyrażoną liczbą próbek, na podstawie których wyznaczana jest prędkość, y_i zmierzoną wartością przemieszczenia w i -tej próbce, zaś x_i chwilą i -tej próbki. Równanie macierzowe

$$\begin{bmatrix} \langle x, x \rangle & \langle 1, x \rangle \\ \langle 1, x \rangle & \langle 1, 1 \rangle \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} \langle x, y \rangle \\ \langle 1, y \rangle \end{bmatrix}, \quad (2.7)$$

opisujące współczynniki prostej

$$y = c_1x + c_2, \quad (2.8)$$

można uprościć przy założeniu, że otrzymywane pomiary przebytej drogi są wykonywane w stałym interwale czasowym. Uproszczenie równań powoduje, że działania na skończonej reprezentacji liczb nie wprowadzają dużych błędów numerycznych. Uproszczone składowe równania (2.7) zapisujemy jako

$$\langle x, x \rangle = \sum_{i=1}^m x_i^2 = \sum_{i=1}^m (x_1 + (i-1)\tau)^2 = \tau^2 \sum_{i=1}^{m-1} i^2, \quad x_1 = 0, \quad (2.9)$$

$$\langle 1, x \rangle = \sum_{i=1}^m x_i = mx_1 + \sum_{i=1}^{m-1} i\tau = \sum_{i=1}^{m-1} i\tau = \frac{m^2 - m}{2}\tau, \quad x_1 = 0, \quad (2.10)$$

$$\langle 1, 1 \rangle = m, \quad (2.11)$$

$$\langle 1, y \rangle = \sum_{i=1}^m y_i, \quad (2.12)$$

$$\langle x, y \rangle = \sum_{i=1}^m x_i y_i = \sum_{i=1}^m (x_1 + (i-1)\tau)y_i = \tau \sum_{i=1}^{m-1} i y_{i+1}, \quad x_1 = 0, \quad (2.13)$$

gdzie τ jest interwałem czasu pomiaru przemieszczenia kątownego koła. Prędkość ω można zatem przedstawić w postaci równania

$$\omega = c_1 = \frac{m\langle x, y \rangle - \langle 1, x \rangle \langle 1, y \rangle}{m\langle x, x \rangle - \langle 1, x \rangle^2}. \quad (2.14)$$

2.5 Regulatory prędkości obrotowej kół

Zadaniem regulatora prędkości obrotowej kół jest dostosowanie prędkości kół do wartości prędkości zadanych. Regulatory prędkości obrotowej kół zostały zaprogramowane jako regulatory PI. Takie rozwiązanie wybrano ze względu na szybką reakcję silnika na sygnał wyjściowy regulatora, co sprawia, że człon różniczkujący wpływa w minimalnym stopniu na proces sterowania i może być pominięty. Regulator został zaprogramowany z wykorzystaniem postaci *IND*, a całkowanie wykonywane jest za pomocą metody trapezów [4]. Prawo regulacji ma postać

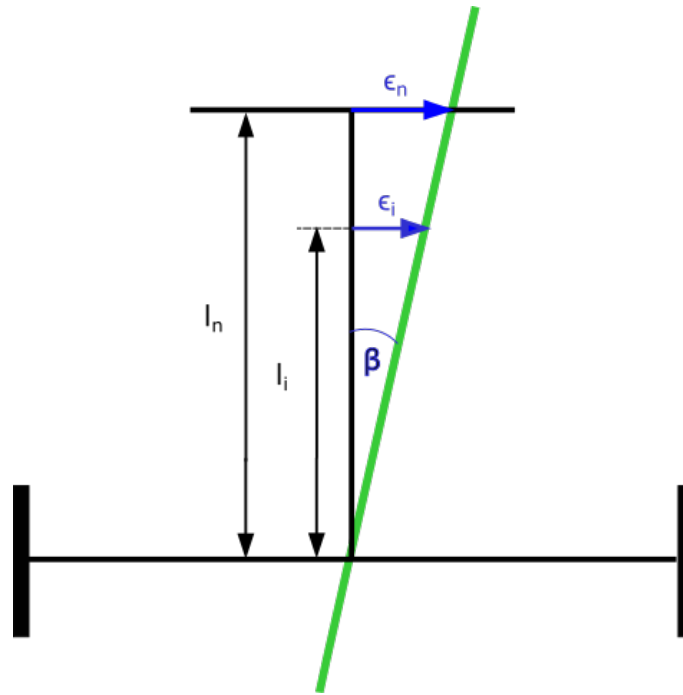
$$y(t) = k_p \epsilon(t) + k_i \int_0^t \epsilon(\tau) d\tau, \quad (2.15)$$

gdzie współczynnik k_p jest wzmocnieniem członu proporcjonalnego, k_i wzmocnieniem członu całkującego, $\epsilon(t)$ uchybem regulacji w chwili t , a $y(t)$ wyjściem regulatora.

2.6 Regulator położenia względem linii

W robotach klasy linefollower kontrolę położenia względem linii najczęściej implementuje się jako regulator PID, który również został zastosowany w tej konstrukcji. Ogólna postać regulatora jest następująca:

$$y(t) = k_p \epsilon(t) + k_d \frac{d}{dt} \epsilon(t) + k_i \int_0^t \epsilon(\tau) d\tau, \quad (2.16)$$



Rysunek 2.4 Zależność kąta od położenia linii i wysunięcia czujników

gdzie k_p , k_i , k_d są odpowiednio wzmocnieniami członów: proporcjonalnego, całkującego, różniczkującego, zaś $y(t)$ jest wyjściem regulatora, a $\epsilon(t)$ uchybem regulacji.

Robot wykonujący przejazd mapujący porusza się z bardzo małą prędkością, co sprawia, że korzystne jest zastosowanie członu całkującego [9].

2.6.1 Modyfikacja regulatora dla powolnych przejazdów

Specyfika zadania mapowania oraz możliwość modyfikacji położenia czujników zmienia wymagania odnośnie procesu sterowania robotem. Możliwość zmiany położenia czujników linii powoduje, że nastawy regulatora muszą się zmieniać wraz ze zmianą wysunięcia listwy sensorycznej.

Zastosowanie standardowego regulatora PID kontrolującego położenie względem linii jest jednak niewystarczające. Nastawy dobrane dla jednej konfiguracji czujników nie sprawdzają się dla innych, powodując oscylacje robota wokół linii. Nastawy można dobierać empirycznie dla każdej konfiguracji lub uzależnić je od wysunięcia listwy sensorycznej.

Problem został rozwiązany poprzez uzależnienie nastaw regulatora PID od wartości wysunięcia czujników względem środka osi robota. Wraz ze zmniejszeniem wysunięcia zmniejsza się czas na reakcję regulatora na zachodzące zmiany, co sprawia, że konieczne jest odpowiednie zwiększenie reakcji wyjścia regulatora na wartość uchybu regulacji. Szybkość reakcji regulatora jest niezwykle istotna podczas przejazdu przez zakręt pod kątem prostym, ponieważ taki przejazd jest szczególnym przypadkiem, w którym ważny jest czas reakcji robota.

Modyfikacja polega na zmianie wzmocnienia członu proporcjonalnego tak, aby odchylenie o ten sam kąt β powodowało zbliżoną reakcję regulatora (rysunek 2.4). Człon całkujący przyczynia się do powolnego dochodzenia do stanu równowagi, a człon różniczkujący zwiększa szybkość stabilizacji, co sprawia, że wartości ich

wzmocnień nie wymagają zmiany wraz ze zmianą położenia czujników linii. Modyfikacje regulatora opisanego równaniem (2.16) dokonano zakładając, że składowa członu proporcjonalnego $y_p(t)$

$$y_p(t) = k_p \epsilon(t), \quad (2.17)$$

ma przyjmować takie same wartości dla takiego samego kąta wychylenia robota. Pomiedzy wysunięciem czujników a położeniem linii względem środka listwy sensorycznej istnieje zależność

$$\frac{\epsilon_n(t)}{\epsilon_i(t)} = \frac{l_n}{l_i}, \quad i = 1 \dots n - 1, \quad (2.18)$$

gdzie $\epsilon_i(t)$ jest i -tym uchybem regulacji, l_i wartością wysunięcia w i -tej konfiguracji, a n liczbą dopuszczalnych konfiguracji. Wykorzystując związek (2.18) zapisujemy równanie (2.17) dla i -tej konfiguracji

$$y_p(t) = \frac{l_n}{l_i} k_p \epsilon_i(t), \quad (2.19)$$

a następnie zapisujemy pełne równanie regulatora w postaci

$$y(t) = \frac{l_n}{l_i} k_p \epsilon_i(t) + k_d \frac{d}{dt} \epsilon_i(t) + k_i \int_0^t \epsilon_i(\tau) d\tau. \quad (2.20)$$

Postać regulatora (2.20) sprawdza się jednak tylko podczas wolnych przejazdów mapujących, a przejazdy z dużymi prędkościami wymagają indywidualnego doboru nastaw dla każdej konfiguracji czujników.

Rozdział 3

Mapowanie trasy

Optymalizacja przejazdu robota wymaga wiedzy na temat przebiegu trasy z wyprzedzeniem. Znajomość przebiegu trasy pozwala na wyznaczenie optymalnej trajektorii, po której porusza się robot. Wiedzę na temat trasy pozyskuje się poprzez wykonanie wstępnego przejazdu mapującego, dzięki któremu robot poznaje trasę. Jakość optymalizacji przejazdu zależy od dokładności mapy tworzonej przez robota, dlatego proces jej tworzenia jest bardzo istotny. Dokładność mapy zależy od błędów pomiarów, których wpływ na sumaryczny błąd określania położenia linii jest różny w zależności od konfiguracji robota. W tym, w zależności od wysunięcia listwy sensorycznej wpływ błędu określania orientacji robota na dokładność wyznaczania mapy zmienia się. Co więcej, możliwość zmiany konfiguracji czujników linii utrudnia zadanie sterowania konstrukcją, co wymaga modyfikacji algorytmu sterowania. Dostosowanie algorytmu sprawia, że jakość sterowania jest zadowalająca niezależnie od wysunięcia czujników. W rezultacie mapowanie nie jest obciążone błędami wynikającymi z nieodpowiedniego sterowania.

3.1 Reprezentacja trasy i położenia robota

Trasa i położenie robota powinny być reprezentowane w taki sposób, aby proces porównywania wyników, dla różnych konfiguracji robota, był możliwie najprostszy. Różne formy reprezentacji umożliwiają analizę innych parametrów trasy oraz toru ruchu robota, takich jak:

- krzywizna toru,
- długość toru,
- krzywizna przebiegu robota,
- droga pokonana przez robota,
- błąd położenia linii.

Równoważność reprezentacji sprawia, że dane zebrane przez robota nie muszą być duplikowane, a jedyną przechowywaną formą reprezentacji jest krzywizna toru oraz krzywizna przebiegu położenia robota. Ta forma reprezentacji została wybrana, ponieważ algorytmy sterowania oparte na niej są proste i cechują się dużym stopniem niezawodności.

3.1.1 Reprezentacja w postaci współrzędnych kartezyjskich

Reprezentowanie kolejnych punktów trasy i położenia robota w układzie kartezyjskim umożliwia obserwację przebiegu mapy trasy oraz przebiegu położenia robota. Ponadto, jest to forma prosta do interpretacji przez człowieka, co pozwala na wyciąganie wniosków z przeprowadzanych badań oraz szybką diagnostykę poprawności działania algorytmów. Przebieg trasy w postaci punktów określanych przez dwie współrzędne sprawia, że możliwe jest badanie błędu określania położenia linii. Trasa oraz przebieg położenia robota stanowią ciągi punktów. Pojedynczy punkt p_i zapisujemy w postaci

$$p_i = [x_i, y_i]^T, \quad (3.1)$$

gdzie x_i oraz y_i to współrzędne i -tego punktu w układzie kartezyjskim.

3.1.2 Reprezentacja w postaci krzywizny

Reprezentacja w postaci krzywizny pozwala na przetwarzanie danych w sposób automatyczny i autonomiczny. Robot dzięki tej formie reprezentacji jest zdolny do wyznaczenia profilu prędkości. Krzywizna będzie wyliczana dla:

- przebiegu mapy trasy,
- przebiegu położenia robota.

Krzywiznę przebiegu trasy wyznacza się dokonując przekształcenia zbioru punktów $[x_i, y_i]^T$ (3.1) określających położenie linii. Wyznacza się ją na podstawie przyrostów położenia i orientacji między kolejnymi punktami, w następujący sposób:

$$\begin{cases} \Delta x_i = x_i - x_{i-1} \\ \Delta y_i = y_i - y_{i-1} \end{cases}, \quad (3.2)$$

$$\alpha_i = \arctg\left(\frac{dy_i}{dx_i}\right), \quad (3.3)$$

$$\begin{cases} \Delta s_{li} = \sqrt{dx_i^2 + dy_i^2} \\ \Delta \alpha_{li} = \alpha_i - \alpha_{i-1} \end{cases}. \quad (3.4)$$

Obliczone wartości Δs_{li} , $\Delta \alpha_{li}$ to odpowiednio przyrosty położenia i orientacji trasy w i -tym punkcie. Kolejno, obliczone przyrosty podstawiamy do równań

$$\begin{cases} K_{li} = \frac{\Delta \alpha_{li}}{\Delta s_{li}} \\ s_{li} = s_{li-1} + \Delta s_{li} \end{cases}, \quad (3.5)$$

gdzie K_{li} oznacza wyliczone wartości krzywizny trasy, a s_{li} długość trasy do i -tego punktu.

Krzywiznę przebiegu położenia robota wyznacza się korzystając bezpośrednio z chwilowych przyrostów położenia i orientacji robota, obliczanych na podstawie danych z układu odometrii:

$$\begin{cases} T_{ri} = \frac{\Delta s_{Ri} + \Delta s_{Li}}{2} \\ R_{ri} = \frac{\Delta s_{Ri} - \Delta s_{Li}}{L} \end{cases}, \quad (3.6)$$

gdzie T_{ri} , R_{ri} są odpowiednio chwilowymi przemieszczeniami i zmianami orientacji robota w i -tym punkcie, L to odległość między środkami kół robota, zaś Δs_{Ri} , Δs_{Li} są przyrostami drogi przebytej odpowiednio przez prawe i lewe koło.

Analogicznie do krzywizny trasy (3.5), krzywiznę toru ruchu robota K_{ri} oraz przebytą drogę s_{ri} wyznaczamy ze wzoru

$$\begin{cases} K_{ri} = \frac{R_{ri}}{T_{ri}} \\ s_{ri} = s_{ri-1} + T_{ri} \end{cases}, \quad (3.7)$$

gdzie s_{ri} jest drogą przebytą przez robota do i -tego punktu. W obu przypadkach zależność krzywizny w funkcji długości trasy $K(s)$ jest uzyskiwana poprzez uzależnienie wyliczonych z równań (3.5) oraz (3.7) wartości K_i od s_i .

3.2 Budowa mapy trasy

Budowa mapy trasy polega na wyznaczaniu położenia linii na podstawie znajomości parametrów robota, jego położenia oraz położenia linii względem listwy zawierającej czujniki linii. Proces mapowania wymaga przejazdu z niewielką prędkością, która powinna być utrzymywana na stałym poziomie. Mapowanie dokonywane jest poprzez cykliczne wykonywanie następującej procedury:

- wyznaczanie położenia i orientacji robota na podstawie odometrii,
- wyznaczanie położenia linii,
- reprezentacja trasy w postaci krzywizny,
- magazynowanie.

Wyznaczanie położenia robota wymaga informacji o chwilowej zmianie jego położenia i orientacji, które są wyliczane na podstawie równań (3.6). Kolejne położenia robota q_{ri} obliczane są na podstawie równania

$$q_{ri} = \begin{bmatrix} x_{ri} \\ y_{ri} \\ \theta_{ri} \end{bmatrix} = \begin{bmatrix} x_{ri-1} + T_{ri} \cos \theta_{ri} \\ y_{ri-1} + T_{ri} \sin \theta_{ri} \\ \theta_{ri-1} + R_{ri} \end{bmatrix}. \quad (3.8)$$

Następnym etapem jest wyznaczanie współrzędnych położenia linii q_{li} ze wzoru

$$q_{li} = \begin{bmatrix} x_{li} \\ y_{li} \\ \theta_{li} \end{bmatrix} = \begin{bmatrix} x_{ri} + l \cos \theta_{ri} - h_i \sin \theta_{ri} \\ y_{ri} + l \sin \theta_{ri} + h_i \cos \theta_{ri} \\ 0 \end{bmatrix}, \quad (3.9)$$

gdzie l jest wartością wysunięcia przedniej listwy, a h_i i -tym położeniem linii względem środka listwy sensorycznej.

Wyliczone w ten sposób współrzędne położenia robota oraz punktów linii są transformowane do postaci krzywizny. Każdy z przebiegów zapisywany jest w pamięci robota, co umożliwia wykorzystanie ich do dalszego przetwarzania bądź transmisji do zewnętrznego urządzenia.

3.2.1 Analiza błędów mapowania

Dokładność z jaką wyznaczane są kolejne punkty toru zależy między innymi od wartości wysunięcia listwy z czujnikami. Błędy określania położenia robota oraz toru mogą być błędami systematycznymi oraz błędami niesystematycznymi [3]. Błąd określania orientacji jest błędem niesystematycznym, rosnącym wraz z przebytą odległością, dlatego istotna jest redukcja jego wpływu na wyznaczanie położenia kolejnych punktów trasy. Wpływ błędu określania orientacji na dokładność wyznaczania kolejnych punktów toru można zredukować poprzez zmienienie wysunięcia listwy.

Błędy systematyczne są błędami wynikającymi z braku możliwości dokonania idealnego pomiaru wymiarów robota, takich jak średnica koła czy odległość między środkami kół, ale również mogą wynikać z ograniczonej rozdzielczości enkoderów. Ten rodzaj błędu można zredukować dbając o to, żeby niepewności z jakimi dokonywane są pomiary były możliwie bardzo małe. Pomimo dużej dokładności pomiarowej wymiarów robota oraz dużej rozdzielczości enkoderów błędy systematyczne mogą odgrywać istotną rolę, zależnie od konfiguracji robota.

Poprzez analizę niepewności pomiarowych można pokazać wpływ poszczególnych błędów pomiaru na błąd położenia punktów toru. Współrzędne toru są wyznaczane metodą pomiaru pośredniego [15], a więc zależą od niepewności pomiarowej innych wielkości. Obliczenie niepewności pomiarowych z jakimi wyznaczone są kolejne punkty toru Δq_{li} wymaga zastosowania metody różniczki zupełnej [15], zakładając że wielkości potrzebne do obliczenia położenia punktu toru są wzajemnie nieskorelowane. Niepewność wyrażana za pomocą tej metody na podstawie równania (3.9) ma wartość

$$\Delta q_{li} = \left| \frac{\partial q_{li}}{\partial x_{ri}} \right| \Delta x_r + \left| \frac{\partial q_{li}}{\partial y_{ri}} \right| \Delta y_r + \left| \frac{\partial q_{li}}{\partial \theta_{ri}} \right| \Delta \theta + \left| \frac{\partial q_{li}}{\partial h_i} \right| \Delta h, \quad (3.10)$$

gdzie Δq_{li} jest wektorem niepewności i -tych współrzędnych położenia punktu trasy, h_i i -tym pomiarem położenia linii względem środka listwy sensorycznej, Δx_r i Δy_r niepewnościami współrzędnych położenia robota, a $\Delta \theta$, Δh odpowiednio niepewnością orientacji robota i niepewnością położenia linii.

Wyrażenie (3.10) można zapisać jako

$$\Delta q_{li} = \begin{bmatrix} \Delta x_r + |h_i \cos \theta_{ri} + l \sin \theta_{ri}| \Delta \theta + |\sin \theta_{ri}| \Delta h \\ \Delta y_r + |h_i \sin \theta_{ri} - l \cos \theta_{ri}| \Delta \theta + |\cos \theta_{ri}| \Delta h \\ 0 \end{bmatrix}, \quad (3.11)$$

co ma postać

$$\Delta q_{li} = \Delta q_{pi} + \Delta q_{tri}, \quad (3.12)$$

gdzie $\Delta q_{pi} = (\Delta x_{ri} \ \Delta y_{ri} \ 0)^T$ jest wektorem niepewności współrzędnych położenia robota, a Δq_{tri} i -tym wektorem niepewności położenia linii względem robota.

Zakładamy, że regulator PID utrzymujący robota na linii działa idealnie, co oznacza, że linia znajduje się zawsze na środku listwy zawierającej czujniki

$$\forall i \in \mathbb{N} \quad h_i = 0. \quad (3.13)$$

Na podstawie założenia (3.13) zapisujemy

$$\Delta q_{tri} = \begin{bmatrix} |l \sin \theta_{ri}| \Delta \theta + |\sin \theta_{ri}| \Delta h \\ |l \cos \theta_{ri}| \Delta \theta + |\cos \theta_{ri}| \Delta h \\ 0 \end{bmatrix}. \quad (3.14)$$

Następnie wyliczamy długość wektora niepewności położenia punktu toru względem robota ξ_{lri}

$$\xi_{lri} = |\Delta q_{lri}| = |l\Delta\theta + \Delta h|. \quad (3.15)$$

Warto zauważyć, że wartość długości wektora ξ_{lri} jest taka sama dla wszystkich i . Długość wektora niepewności traktujemy jako promień koła o środku w punkcie, w którym w danej chwili znajduje się wyznaczany punkt toru. W tak otrzymanym kole musi znajdować się rzeczywiste położenie punktu. Długości wektorów niepewności spełniają nierówność

$$\xi_{li} \leq \xi_{ri} + \xi_{lri}, \quad (3.16)$$

gdzie ξ_{li} , ξ_{ri} to odpowiednio długość wektora niepewności położenia punktu toru oraz długość wektora niepewności położenia robota.

Założymy, że ξ_{li} przyjmuje największą możliwą wartość, to znaczy

$$\xi_{li} = \xi_{ri} + \xi_{lri} \quad (3.17)$$

i skąd

$$\xi_{li} = |\Delta q_{pi}| + |l\Delta\theta + \Delta h|. \quad (3.18)$$

Długość wektora ξ_{li} jest zależna od współrzędnych wektora Δq_{pi} , co sprawia, że wektor Δq_{pi} musiałby być znany z założenia. Wyliczamy więc niepewność określania położenia i orientacji robota Δq_{ri} na podstawie różniczki zupełnej w następujący sposób

$$\Delta q_{ri} = \sum_{k=1}^i \left| \frac{\partial q_{ri}}{\partial T_{rk}} \right| \Delta T_r + \sum_{k=1}^i \left| \frac{\partial q_{ri}}{\partial \theta_{rk}} \right| \Delta \theta, \quad (3.19)$$

gdzie ΔT_r jest niepewnością przyrostu położenia robota. Wyliczając składowe równania (3.19) otrzymujemy

$$\Delta q_{ri} = \begin{bmatrix} \Delta T_r \sum_{k=1}^i |\cos(\theta_{rk})| + \Delta \theta \sum_{k=1}^i |T_{rk} \sin(\theta_{rk})| \\ \Delta T_r \sum_{k=1}^i |\sin(\theta_{rk})| + \Delta \theta \sum_{k=1}^i |T_{rk} \cos(\theta_{rk})| \\ \Delta \theta \end{bmatrix}. \quad (3.20)$$

Następnie wyliczamy długość wektora Δq_{ri}

$$\begin{aligned} \xi_{ri} = |\Delta q_{ri}| = & \left(\left(\Delta T_r \sum_{k=1}^i |\cos(\theta_{rk})| + \Delta \theta \sum_{k=1}^i |T_{rk} \sin(\theta_{rk})| \right)^2 + \right. \\ & \left. + \left(\Delta T_r \sum_{k=1}^i |\sin(\theta_{rk})| + \Delta \theta \sum_{k=1}^i |T_{rk} \cos(\theta_{rk})| \right)^2 + \Delta \theta^2 \right)^{\frac{1}{2}}. \end{aligned} \quad (3.21)$$

Ponieważ

$$\Delta q_{pi} = \begin{bmatrix} \Delta q_{ri1} \\ \Delta q_{ri2} \\ 0 \end{bmatrix}, \quad (3.22)$$

gdzie Δq_{ri1} , Δq_{ri2} to odpowiednio 1. i 2. składowa wektora Δq_{ri} mamy

$$|\Delta q_{pi}| \leq |\Delta q_{ri}|. \quad (3.23)$$

Z równania (3.17) wynika, że i -ta długość wektora niepewności położenia linii wynosi

$$\xi_{li} = \xi_{ri} + |l\Delta\theta + \Delta h|. \quad (3.24)$$

Równanie (3.24) zawiera dwie składowe. Pierwszą z nich jest długość wektora niepewności określania położenia robota ξ_{ri} , która rośnie wraz ze wzrostem liczby przemieszczeń i zmian orientacji robota. Wzrost ξ_{ri} powoduje ustawiczny przyrost długości wektora niepewności położenia linii ξ_{li} , wraz z przyrostem drogi przebytej przez robota. Jest to spowodowane kumulacją błędu wynikającego z błędów określania orientacji i przemieszczenia. Drugą składową równania (3.24) jest długość wektora niepewności określania położenia punktu toru względem robota ξ_{lr} , na którą składa się niepewność określania orientacji $\Delta\theta$ oraz niepewność położenia linii względem listwy sensorycznej Δh . O poziomie udziału niepewności orientacji w wartości ξ_{lri} decyduje wartość wysunięcia listwy sensorycznej. W efekcie błąd określania orientacji odgrywa większą rolę dla większych wysunięć. Wartość ξ_{lri} jest również zależna od niepewności Δh , która jest zawsze stała i nie powoduje ciągłych przyrostów wartości ξ_{li} wraz z przejechanym dystansem. Wpływ niepewności składowych na długość wektora niepewności określania położenia punktów toru, świadczy o wpływie błędów cząstkowych na błąd mapowania toru. Wpływ błędów pomiaru na proces mapowania w zależności od konfiguracji czujników przedstawiono w podrozdziale 5.2, gdzie zostały zaprezentowane wyniki eksperymentalne.

Rozdział 4

Konfiguracja czujników w procesie sterowania

Proces sterowania robota wykorzystujący algorytmy optymalizacji przejazdu jest wrażliwy na niedokładności wyznaczonej mapy. Własności wyznaczonej mapy zmieniają się wraz z wysunięciem listwy z czujnikami, dlatego dokładność mapy można rozpatrywać według różnych kryteriów, a stopień ich spełnienia może przekładać się na poprawę różnych aspektów optymalizacji przejazdu.

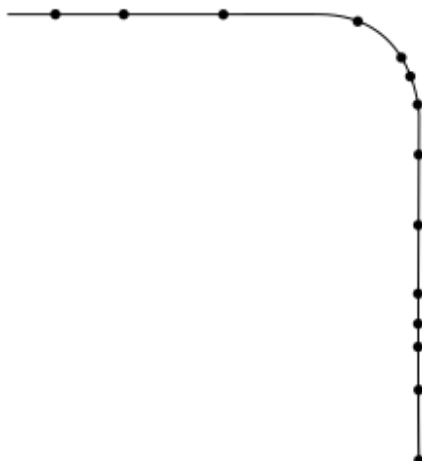
Wyznaczanie profilu prędkości na podstawie krzywizny przebiegu położenia robota również jest zależne od konfiguracji czujników linii, ponieważ wraz ze zmianą wysunięcia zmienia się tor ruchu robota, od którego krzywizny zależą maksymalne dopuszczalne prędkości na zakrętach.

4.1 Wpływ dokładności wyznaczania trasy na poprawność optymalizacji przejazdu

Dokładność wyznaczania trasy odgrywa dużą rolę w optymalizacji przejazdu, ponieważ zadanie optymalizacji wymaga wiedzy o przebiegu trasy zanim robot rozpocznie zoptymalizowany przejazd. Błędy położenia jakimi jest obciążona trasa przekładają się na jakość optymalizacji trasy w zależności od swojej wielkości i charakteru. Jakość wyznaczania trasy można rozpatrywać względem różnych kryteriów, które przekładają się na poprawę działania różnych algorytmów optymalizacji trasy. Poniżej zostaną omówione dwa rodzaje błędów o różnej naturze oraz ich wpływ na optymalizację przejazdu. Błędy wyznaczania położenia trasy można podzielić na:

- błędy nieliniowej parametryzacji czasem,
- błędy deformacji toru.

Błąd nieliniowej parametryzacji czasem powoduje, że długość mapy trasy w danej chwili nie odpowiada rzeczywistej długości linii. Rozbieżności w pomiarze długości trasy wynikają z nieliniowej parametryzacji przyrostu drogi, a nieliniowość ta ukazuje się w różnym stopniu dla różnych wysunięć listwy sensorycznej. Przejazd robota można podzielić na zdarzenia, które występują w równym interwale czasu. Przykładowe miejsca, w których występują zdarzenia na mapie trasy przedstawia rysunek 4.1. W ogólności, kolejne zdarzenia występują względem siebie w różnych



Rysunek 4.1 Trasa sparametryzowana czasem

odległościach, co powoduje błąd określania położenia względem przejazdu idealnego, w którym zdarzenia byłyby rozłożone równomiernie (rysunek 4.2).

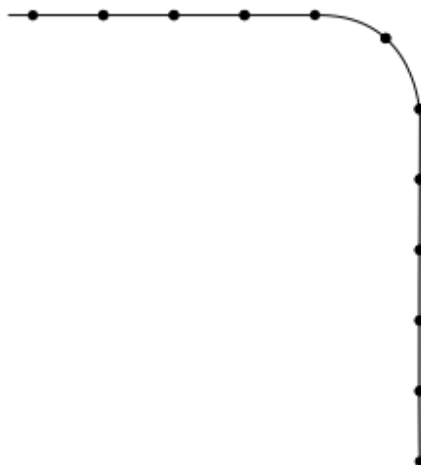
Błąd określania położenia linii wynikający z nieliniowości parametryzacji pozwala na odtworzenie kształtu przebiegu trasy z idealną dokładnością. Kształt mapy trasy dobrze odwzorowujący rzeczywisty kształt linii, przyczynia się do poprawnego działania algorytmów optymalizacji. Ograniczeniem spowodowanym przez nieliniową parametryzację jest możliwość poprawnej samolokalizacji robota na trasie, która jest możliwa tylko przy takiej samej parametryzacji lub po jej skorygowaniu.

Błąd deformacji toru wynika ze zmiany krzywizny toru robota. Długość toru między kolejnymi zdarzeniami jest zachowana i taka sama jak podczas przejazdu idealnego, natomiast odległość między odpowiadającymi sobie zdarzeniami obu przebiegów jest różna od zera (rysunek 4.3). Błąd powodujący odkształcenie toru robota przyczynia się do zmniejszenia jakości optymalizacji przejazdu, ponieważ krzywizna mapy trasy nie odzwierciedla rzeczywistej krzywizny. Niedokładne odwzorowanie kształtu toru może powodować problemy z poprawną optymalizacją toru ruchu robota.

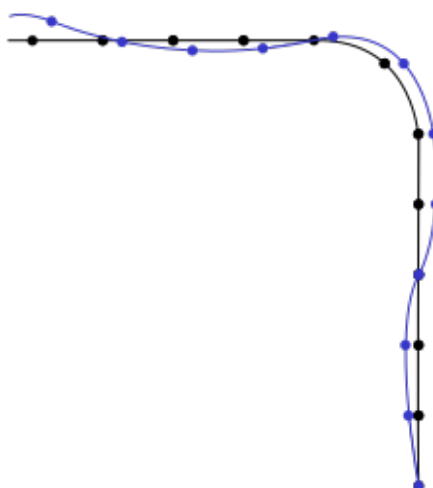
Tworzone mapy nigdy nie są idealne i zawsze są obarczone błędami wynikającymi z błędów pomiaru chwilowych przemieszczeń i orientacji. Błąd parametryzacji w najlepszym przypadku można przyjąć za liniowy, a błąd deformacji za bardzo mały. Zwiększanie się błędu wyznaczania położenia linii wraz ze wzrostem wysunięcia przekłada się na sumę tych błędów. Proporcja między błędem parametryzacji a błędem deformacji jest zależna od sposobu w jaki porusza się robot, a ten z kolei od wysunięcia listwy sensorycznej.

4.2 Wyznaczanie profilu prędkości w zależności od konfiguracji

Wyznaczanie profilu prędkości dokonuje się w celu ograniczenia prędkości z jaką może poruszać się robot na zakrętach, tak aby nie wpadał w poślizg. Dozwolone



Rysunek 4.2 Parametryzacja idealna trasy



Rysunek 4.3 Zestawienie trasy idealnej i zdeformowanej

prędkości są dobierane w zależności od krzywizny toru robota oraz dozwolonych przyspieszeń. Profile prędkości ulegają zmianie wraz ze zmianą pozycji czujników, która wpływa na tor ruchu robota.

Robot z czujnikami umieszczonymi w niewielkiej odległości od swojego środka przemieszcza się niemal idealnie nad linią definiującą tor. Kształt przebiegu położeń robota jest praktycznie taki sam jak kształt toru. Podobieństwo między przebiegiem kolejnych położeń robota a przebiegiem trasy sprawia, że krzywizna toru ruchu robota jest mniejsza niż dla przypadku, gdy listwa jest wysunięta na większą odległość. Wyjątkiem jest zakręt w postaci kąta prostego, gdzie wartość wysunięcia listwy wpływa na wygładzanie zakrętu.

Wyznaczanie profilu prędkości odbywa się w następujących etapach:

- wygładzanie przebiegu krzywizny,
- wyznaczanie prędkości maksymalnych,
- uwzględnienie ograniczeń przyspieszenia wzdłużnego.

Krzywizna toru ruchu robota jest obliczana na podstawie pomiarów wykonywanych w krótkim interwale czasu, co powoduje, że pomiary są zaszumione. W celu wyznaczenia profilu prędkości należy najpierw odfiltrować przebieg stosując np. metodę średniej ruchomej [1], która ułatwia wyznaczanie maksymalnej dozwolonej prędkości na zakrętach oraz detekcję zakrętów. Maksymalna prędkość na zakrętach jest wyznaczana globalnie, to znaczy, że maksymalna krzywizna podczas całego przejazdu robota determinuje maksymalną prędkość na wszystkich zakrętach $v_{z,max}$. Maksymalną prędkość wyznaczamy w następujący sposób:

$$v_{z,max} = \sqrt{\frac{a_{d,max}}{|\max(K_r(s_{ri}))|}}, \quad (4.1)$$

gdzie $K_r(s_{ri})$ jest krzywizną przebiegu położeń robota w funkcji drogi, a $a_{d,max}$ maksymalnym osiągalnym przyspieszeniem dośrodkowym, którego wartość wynika z maksymalnego tarcia poprzecznego kół.

Prędkość maksymalna na odcinkach prostych wynika z ograniczeń konstrukcyjnych. W przypadku przedstawionej w tej pracy konstrukcji górne ograniczenie prędkości jest spowodowane zjawiskiem aliasingu [19], które wynika ze zbyt dużej liczby impulsów sygnału kwadraturowego w jednostce czasu. Zjawisko to powoduje gwałtowne zmniejszenie wartości mierzonej prędkości, mimo wzrostu jej rzeczywistej wartości.

Profil prędkości przyjmujący tylko dwie różne wartości, wymagałby możliwości rozpędzania robota w nieskończenie krótkim czasie, czyli przyspieszenia wzdłużne musiałyby mieć nieskończoną wartość. Przyspieszenia z jakimi może poruszać się robot są jednak ograniczone głównie z powodu maksymalnej wartości tarcia wzdłużnego, przy której robot nie wpada w poślizg. Ograniczenie to powoduje, że prędkość robota v musi się zmieniać tak, aby spełniony był warunek

$$|v - v_p| < a_{w,max} \Delta t, \quad (4.2)$$

gdzie v_p jest prędkością początkową robota, $a_{w,max}$ maksymalnym dopuszczalnym przyspieszeniem wzdłużnym, a Δt przedziałem czasu, w którym zachodzi zmiana prędkości.

Rozdział 5

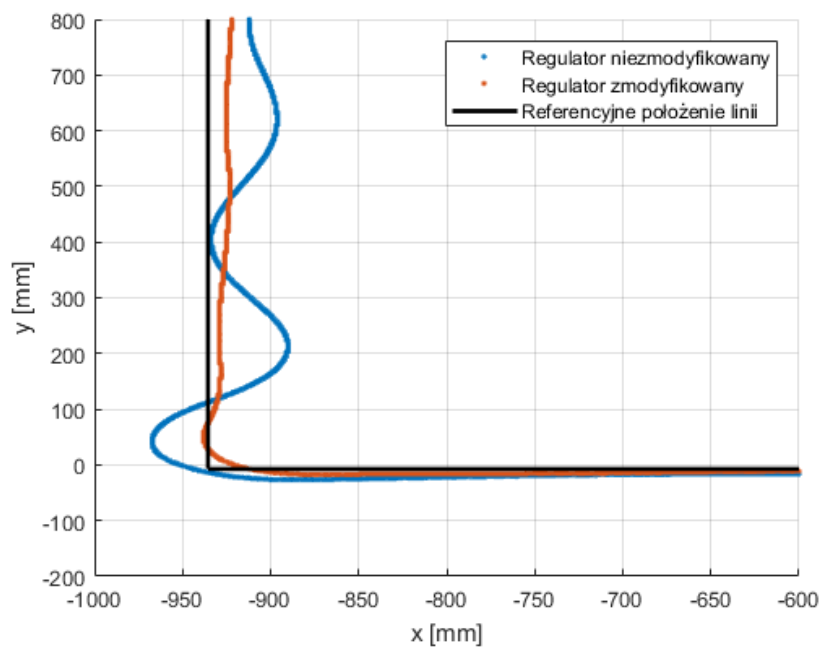
Badania eksperymentalne

Badania eksperymentalne mają na celu sprawdzenie potencjalnych korzyści mogących wynikać z zastosowania różnych wysunięć listwy z czujnikami. Dokonano serii eksperymentów dla różnych wysunięć listwy sensorycznej, a następnie wykreślono zależności pokazujące zmiany błędów mapowania. Przeprowadzono badania testujące stopień dopasowania kształtu mapy trasy, a ich wyniki mogą posłużyć do usprawnienia procesów optymalizacji. Rozpatrzono również jaki jest wpływ konfiguracji czujników na wyznaczanie profilu prędkości robota.

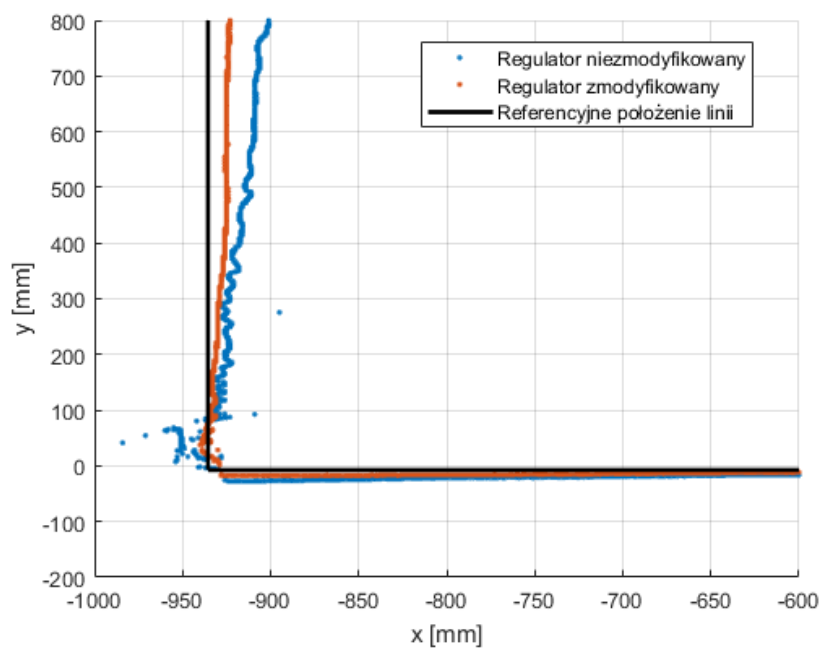
5.1 Wpływ modyfikacji regulatora PID na dokładność wyznaczania położenia linii

Modyfikacja regulatora przedstawiona w podrozdziale 2.6.1 sprawia, że nastawy dla różnych wysunięć listwy z czujnikami są odpowiednio dobrane. Przedstawiono rezultaty modyfikacji regulatora, które pokazują wpływ modyfikacji na tor ruchu robota oraz przebieg mapy trasy.

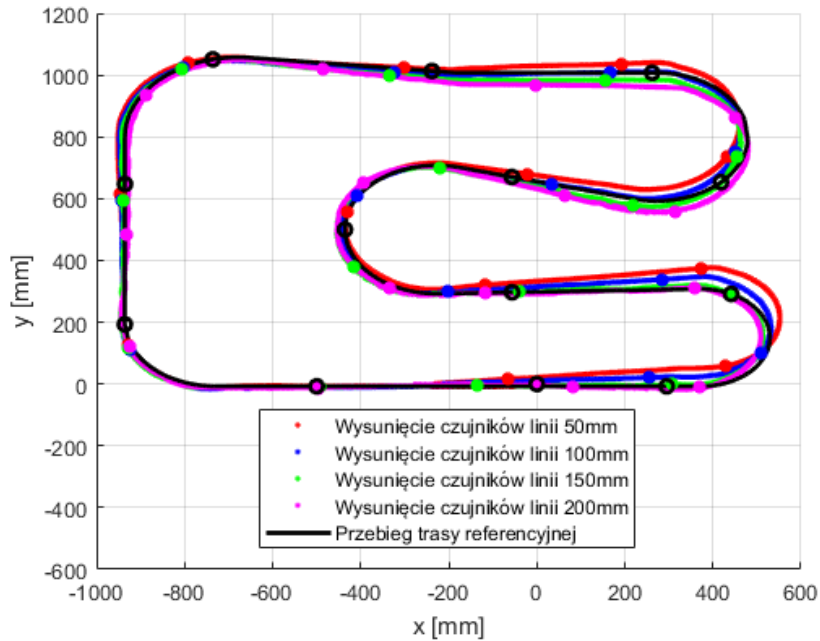
W analizowanym przypadku nastawy regulatora położenia względem linii zostały dobrane w sposób empiryczny przy wysunięciu $200mm$. Zmniejszenie wysunięcia przyczynia się do skrócenia czasu, w którym sterownik może zareagować na zmiany. Nieodpowiednie sterowanie podczas przejazdu przez kąt prosty powoduje oscylacje przebiegu położeń robota względem linii, a one z kolei wpływają na błąd określania położenia linii. Dostosowanie regulatora poprzez uzależnienie równania (2.16) od wysunięcia czujników, pozwala wyeliminować oscylacje. Zastosowanie nowej postaci regulatora (2.20) sprawia, że robot porusza się, jakby nastawy zostały dobrane indywidualnie dla każdej konfiguracji (rysunek 5.1). Płynność przejazdu przez kąt prosty powoduje, że przebieg trasy uzyskanej w procesie mapowania nie oscyluje. Ponadto, środek listwy sensorycznej porusza się niemal idealnie po linii, co uzasadnia założenie (3.13). Wzrost poprawności mapowania linii przedstawia rysunek 5.2.



Rysunek 5.1 Porównanie toru ruchu robota dla wysunięcia listwy 50mm



Rysunek 5.2 Porównanie uzyskanym map trasy dla wysunięcia listwy 50mm



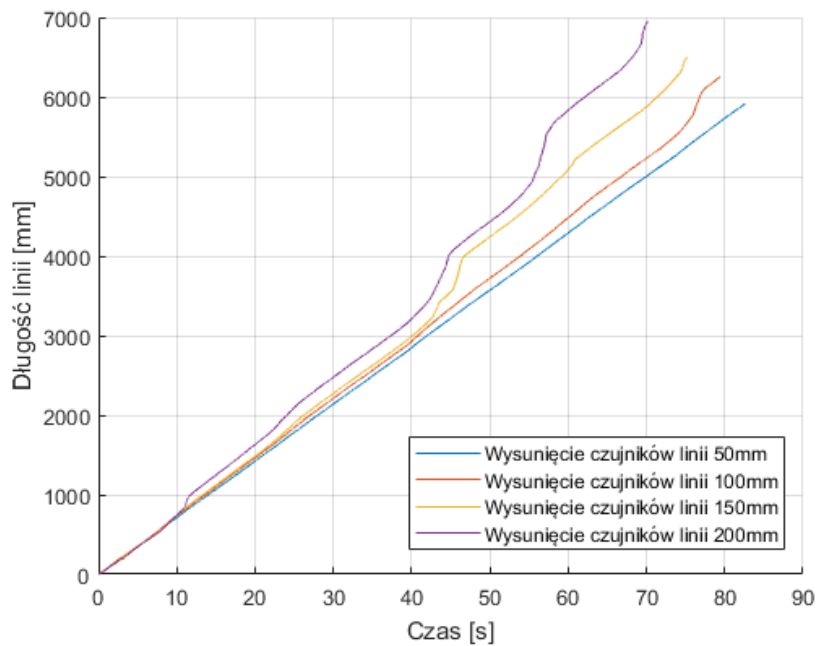
Rysunek 5.3 Zestawienie uzyskanych map tras

5.2 Wpływ konfiguracji czujników na dokładność mapowania

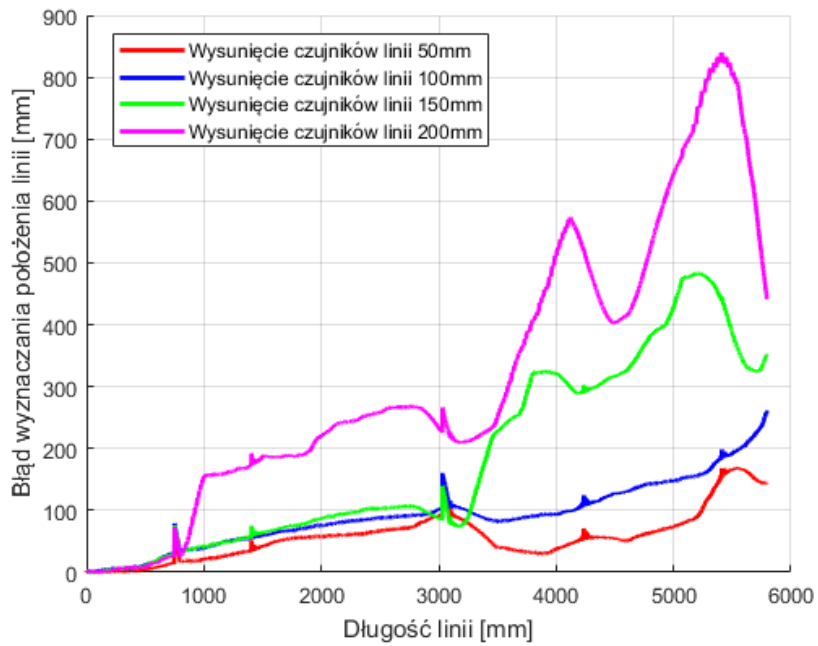
Badanie wpływu wysunięcia modułu czujników poprzez wykonywanie serii eksperymentów, podczas których wyznaczana jest trasa, ma na celu pokazanie zależności między wysunięciem a wielkością błędu mapowania. Błąd wyznaczania trasy może zmieniać swoją wartość i charakter w zależności od trajektorii, po której porusza się robot, a ta z kolei zależy od wysunięcia listwy sensorycznej. Błąd określania położenia linii został zmierzony jako odległość między punktami trasy powstałej w procesie mapowania a punktami trasy referencyjnej, dla których długość linii jest taka sama. Wielkość tego błędu jest w dużym stopniu uzależniona od wysunięcia listwy z czujnikami. Na rysunku 5.3 widzimy, że mapy tras są zdeformowane i w różnym stopniu odwzorowują przebieg trasy referencyjnej. Ponadto znaczniki, którymi oznakowana jest trasa co $0,5m$, obrazują jak błędy mogą powodować zmianę długości mapowanej trasy. Rysunek 5.4 pokazuje, że pomiar długości linii dla różnych konfiguracji jest nieliniowo sparametryzowany czasem.

Błąd mapowania został zmierzony względem trasy referencyjnej i przedstawiony w funkcji długości linii dla różnych wysunięć listwy sensorycznej na rysunku 5.5. Błąd z jakim wyznaczane są kolejne punkty mapy rośnie wraz długością trasy, a jego przyrost jest znacznie większy dla większych wysunięć listwy sensorycznej. Przyrosty błędów na odcinkach prostych są zbliżone, natomiast różnica ukazuje się na zakrętach. Zjawisko to jest spowodowane gwałtownym przyrostem na nich mierzonej długości linii dla dużych wysunięć listwy (zobacz rysunek 5.4).

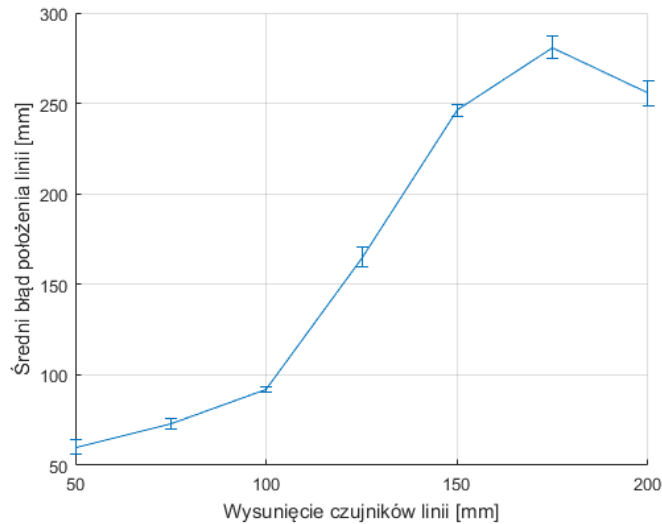
Wykreślona została zależność średniego błędu położenia linii w funkcji wysunięcia modułu czujników (rysunek 5.6). Prezentowane wyniki zostały uzyskane na podstawie siedmiu przejazdów mapujących dla siedmiu konfiguracji modułu czujników. Niepewność wyników została wyznaczona poprzez odchylenie standardowe



Rysunek 5.4 Przyrost długości mapowanej linii w czasie



Rysunek 5.5 Błąd położenia linii dla różnych wysunięć czujników



Rysunek 5.6 Średni błąd położenia linii dla różnych wysunięć czujników

średniej [15].

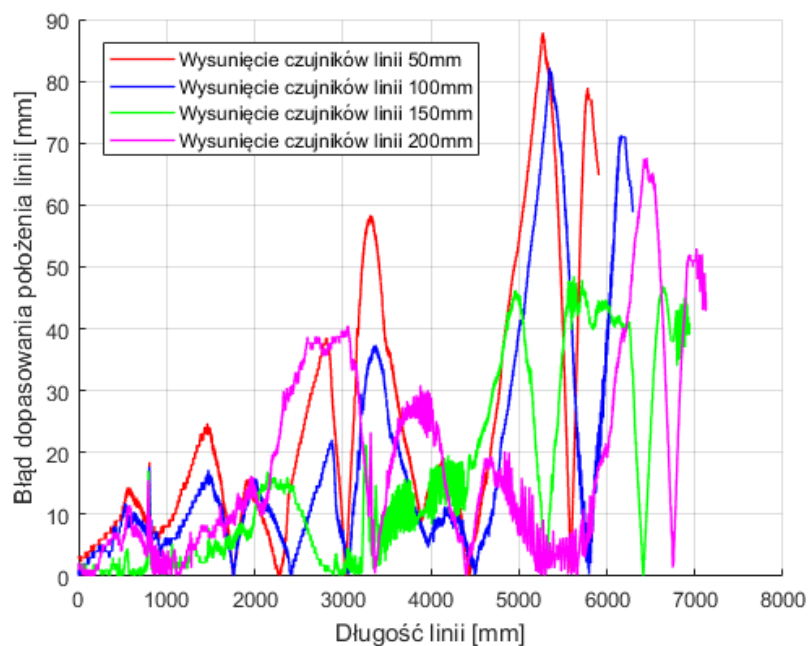
Rysunek 5.6 prezentuje, jak dokładność wyznaczania mapy rośnie wraz ze zmniejszeniem wysunięcia. Na podstawie równania niepewności pomiarowej (3.24) widzimy, że jedyną składową zależną od wysunięcia jest składowa $l\Delta\theta$, która świadczy o wzroście znaczenia błędu określania orientacji wraz ze wzrostem wysunięcia l . Wynika z tego, że błąd określania orientacji robota jest wzmacniany przez wartość wysunięcia listwy z czujnikami.

Dokładność z jaką wyznaczana jest trasa może być również rozpatrywana pod kątem dopasowania kształtu, czyli pomiaru odległości między punktami mapy trasy a najbliższymi odpowiednimi im punktami trasy referencyjnej. Kształt trasy jest odwzorowywany w różnym stopniu, zależnie od tego jak skonfigurowany jest robot. Dokładność dopasowania przebiegu trasy nie jest proporcjonalna do dokładności, z jaką wyznaczane są kolejne położenia punktów trasy. Przebiegi błędów dopasowania dla różnych konfiguracji czujników zostały przedstawione na rysunku 5.7. O jakości dopasowania świadczy to, jak bardzo mapa trasy jest podobna do przebiegu trasy referencyjnej na całej długości. Stąd wartości błędów dopasowania dla różnych konfiguracji zostały uśrednione i przedstawione na rysunku 5.8.

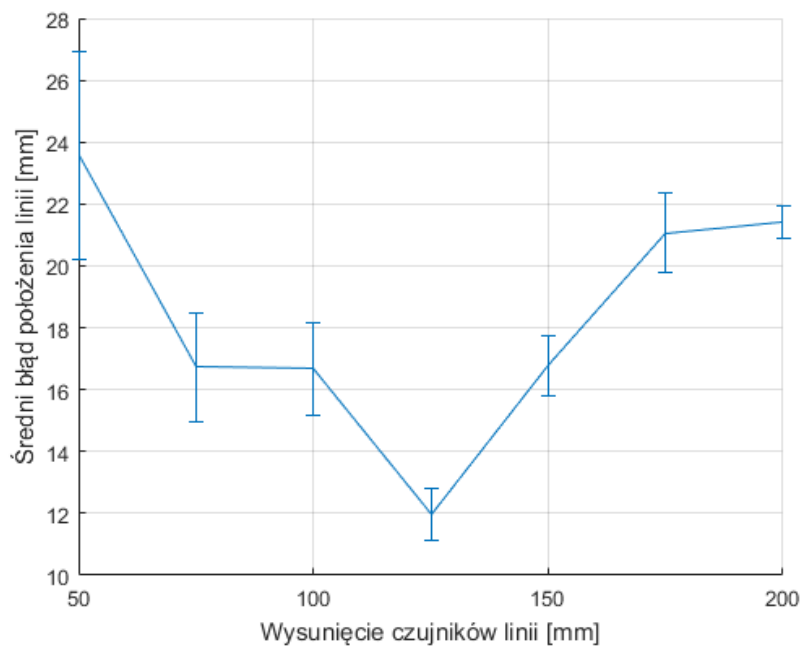
5.3 Analiza wyników mapowania trasy pod kątem optymalizacji przejazdu

Zmiana konfiguracji czujników może powodować zmianę charakteru błędów mapowania trasy. Różne formy przejawiania się błędów mogą wpływać w różny sposób na proces optymalizacji przejazdu. Zmieniając wysunięcia listwy sensorycznej poruszamy się między dwoma przypadkami, w których:

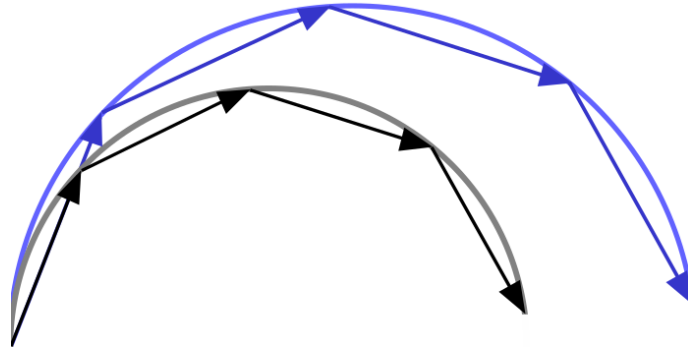
- parametryzacja drogi czasem jest nieliniowa (zwiększenie prędkości na zakrętach), a błędy odwzorowania kształtu są bardzo małe,
- parametryzacja drogi czasem jest niemal linowa, ale występują błędy odwzorowania kształtu na zakrętach.



Rysunek 5.7 Błędy dopasowania linii dla różnych wysunięć czujników



Rysunek 5.8 Średni błąd dopasowania trasy dla różnych wysunięć czujników linii



Rysunek 5.9 Idea zmiany promienia zakrętu w wyniku zmiany wartości translacji

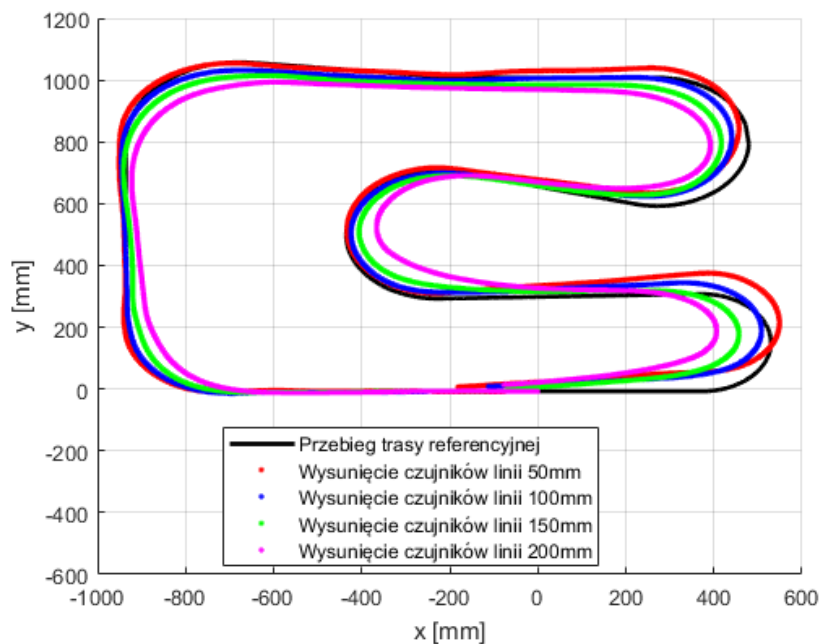
W pewnym zakresie zwiększanie wysunięcia listwy z czujnikami przyczynia się do lepszego odwzorowania kształtu zakrętów, ponieważ błąd mapowania zakrętu zależy głównie od błędu zmiany orientacji robota, która w tym miejscu zmienia się najszybciej. Jednakże dowolne zwiększenie wysunięcia prowadzi do pogorszenia odwzorowania kształtu. Równocześnie, przy dużych wysunięciach listwy błąd mapowania przekłada się w szczególności na nieliniowość parametryzacji czasem na zakrętach (rysunek 5.4). Tym niemniej, parametryzacja czasem nie ogrywa dużej roli w procesie optymalizacji przejazdu, ponieważ szybki przejazd jest zwykle wykonywany przy tym samym wysunięciu czujników linii. Co powoduje, że i przy mapowaniu i w trakcie przejazdu właściwego używamy takiej samej parametryzacji. Taka sama parametryzacja podczas przejazdu mapującego i przejazdu zoptymalizowanego sprawia, że robot jest w stanie dobrze określać swoje położenie względem kolejnych zakrętów.

Brak deformacji przebiegu trasy może wpływać pozytywnie na działanie algorytmów optymalizacji toru ruchu robota [5], ponieważ optymalna ścieżka jest wyznaczana na podstawie kształtu toru. Przebieg trasy dla małych wysunięć jest niemal linowo sparametryzowany czasem, a błąd systematyczny pomiaru translacji przekłada się na zmianę promienia zakrętu. Promień zakrętu ulega przeskalowaniu, ponieważ mierzone przemieszczenie robota nie jest równe jego rzeczywistemu przemieszczeniu (zobacz rysunek 5.9). Błędy mapowania powstają głównie na zakrętach, dlatego można przypuszczać, że przejazd dla małych wysunięć czujników również może być użyteczny z punktu widzenia optymalizacji, jeśli weźmie się pod uwagę poprawkę na odkształcenia zakrętów.

5.4 Profil prędkości robota w zależności od konfiguracji czujników

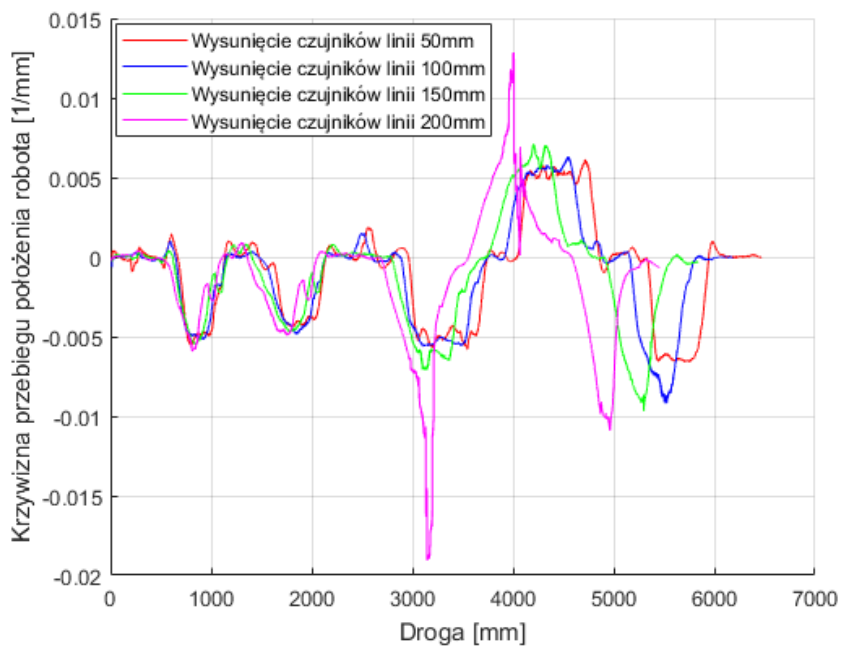
Optymalizacja profilu prędkości dla różnych wysunięć listwy sensorycznej daje różne rezultaty. Sposób poruszania się robota zależy od wysunięcia modułu czujników, co powoduje, że krzywizna toru również zależy od wysunięcia. Zmiana krzywizny toru robota przekłada się na maksymalne dopuszczalne prędkości, z którymi robot może wjeżdżać w zakręty, a to z kolei powoduje różnice między profilami prędkości dla różnych konfiguracji czujników.

Obserwując przebieg położenia robota w układzie kartezjańskim (rysunek 5.10) można zauważyć, że robot z większym wysunięciem modułu czujników wykonuje

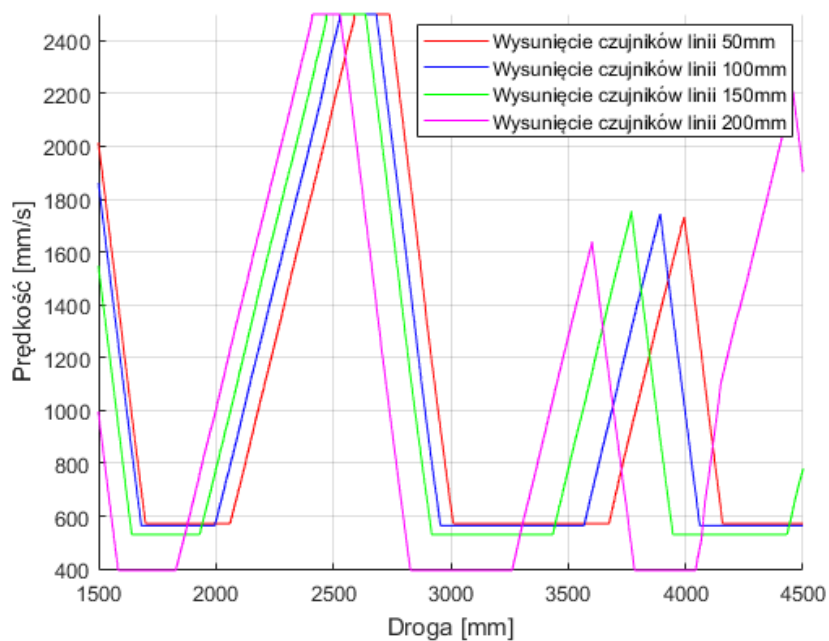


Rysunek 5.10 Zestawienie przebiegów położenia robota dla różnych konfiguracji czujników

skręty o mniejszym promieniu, a droga pokonana przez robota jest krótsza. Na podstawie przebiegów kolejnych położenia robota można wyznaczyć przebiegi krzywizny, a następnie je porównać (rysunek 5.11). Porównując przebiegi krzywizny dla różnych konfiguracji czujników, można zauważyć, że tor poruszania się robota wygląda się dla mniejszych wysunięć listwy sensorycznej. Przebiegi przedstawione na rysunku 5.11 są różnej długości, ponieważ robot z większym wysunięciem czujników porusza się po wewnętrznej stronie zakrętów, skracając w ten sposób drogę, którą musi pokonać. Mniejsza krzywizna toru na zakrętach dla mniejszych wysunięć zwiększa dozwoloną prędkość, z którą robot może poruszać się na zakręcie. W związku ze zmianą dopuszczalnych prędkości profile prędkości również ulegają zmianie (rysunek 5.12).



Rysunek 5.11 Zestawienie krzywizn przebiegów położenia robota dla różnych konfiguracji czujników



Rysunek 5.12 Zestawienie profili prędkości dla różnych konfiguracji czujników

Rozdział 6

Podsumowanie

Celem niniejszej pracy była ocena wpływu konfiguracji czujników linii na poszczególne etapy optymalizacji przejazdu robota klasy linefollower. Optymalizacja przejazdu robota wymaga wykonania próbnego przejazdu mapującego, którego celem jest zebranie informacji o trasie. W pracy wykazano, że mapy budowane przy mniejszych wysunięciach listwy z czujnikami charakteryzują się większą dokładnością. Pokazano też, że większa dokładność wyznaczania trasy niekoniecznie pociąga za sobą większą dokładność odwzorowania kształtu toru. Wykazano, że profilowanie prędkości dla mniejszych wartości wysunięcia listwy z czujnikami daje lepsze rezultaty.

W pracy zaproponowano sposób analizy błędów mapowania poprzez analizę niepewności pomiarowych, który uprościł interpretację przyczyn występowania błędów mapowania oraz potwierdził hipotezę o zależności dokładności mapowanej trasy od wysunięcia listwy sensorycznej. Pokazano w ten sposób, że dokładność mapy trasy może być zwiększona poprzez zmniejszenie wysunięcia listwy z czujnikami.

Przeprowadzono serię eksperymentów, których celem było zbadanie zależności błędu mapowania trasy od konfiguracji czujników z wykorzystaniem rzeczywistego robota. Testy zaimplementowanych algorytmów mapowania trasy dla różnych konfiguracji czujników potwierdziły wnioski wyciągnięte na podstawie analizy niepewności pomiarowych. Ponadto, przeprowadzone badania wykazały, że błąd mapowania może przyczyniać się do zmiany kształtu mapy trasy oraz zmiany jej długości. Wynika z tego, że dokładność z jaką wyznaczana jest mapa, nie jest równoważna dokładności odwzorowania kształtu toru. Dokładność z jaką odwzorowywany jest kształt toru, wynikająca z przeprowadzonych eksperymentów, jest największa dla wartości wysunięcia wynoszącej $125mm$. Można twierdzić, że wykorzystanie wartości wysunięcia, dla której kształt toru jest odwzorowany najlepiej, będzie przyczyniało się do poprawy działania algorytmu optymalizacji toru ruchu robota.

Z przeprowadzonych badań wynika, że wyznaczone profile prędkości robota dla różnych konfiguracji czujników ulegają zmianie. Maksymalne prędkości z jakimi robot może poruszać się na zakrętach bez poślizgu są wyższe przy mniejszych wysunięciach listwy sensorycznej.

Poza problemami opisanymi w pracy napotkano inne, pomniejsze. Jednym z takich problemów zaobserwowanych podczas badań było występowanie luzów w przekładniach silników. Duża średnica kół robota sprawiła, że luzy przekładni stały się kłopotliwe w przypadku zmiany kierunku obrotu kół. Sterownik kontrolujący prędkość obrotową koła nie był w stanie ustabilizować położenia wału silnika, ponieważ ze względu na luzy przekładni przy zmianie kierunku ruchu silnik nie był obciążony.

Problem nie wpływał jednak w sposób znaczący na jakość wyznaczonej mapy, ponieważ błędy odczytu przemieszczenia spowodowane luzami przekładni wzajemnie się kompensują. Co więcej, prędkość maksymalna robota jest ograniczona z powodu zjawiska aliasingu. Ograniczenie to jest jednak akceptowalne przy zastosowanym rozmiarze kół.

W dalszych pracach robot powinien zostać wyposażony w koła o mniejszej średnicy, co za tym idzie, powinien zostać zaimplementowany filtr antyaliasingowy. Kolejną możliwością rozwoju algorytmu sterowania powinna być implementacja algorytmu optymalizacji toru ruchu robota. Wykorzystanie tego algorytmu wraz ze zdobytą wiedzą na podstawie przeprowadzonych eksperymentów mogłoby przyczynić się do poprawy działania algorytmu optymalizacji toru ruchu robota. Warto również rozważyć wykorzystanie układów pomiaru stanu wewnętrznego, w celu zwiększenia dokładności określania kąta, pod którym znajduje się robot.

Bibliografia

- [1] Dokumentacja MathWorks moving mean. <https://www.mathworks.com/help/matlab/ref/movmean.html>.
- [2] Robotic Arena 2018 regulamin konkurencji linefollower light. https://www.roboticarena.pl/media/filer_public/a2/d3/a2d38fd7-7de8-42fb-a59d-6c7361d76e14/lf_light.pdf.
- [3] J. Borenstein, L. Feng. A method for measuring, comparing, and correcting dead-reckoning errors in mobile robots. <http://www-personal.umich.edu/~johannb/Papers/umbmark.pdf>, Gru. 1994.
- [4] S. C. Chapra, R. P. Canale. *Numerical Methods for Engineers*. McGraw-Hill, 2010.
- [5] R. Cymiński. Optymalizacja toru ruchu robota klasy linefollower. Praca magisterska, Politechnika Wrocławska, 2018.
- [6] T. Januszewski. Metody reprezentacji i śledzenia trasy dla robota klasy linefollower. Projekt inżynierski, Politechnika Wrocławska, 2017.
- [7] Kingbright. *KTIR0711S datasheet*, Mar. 2017. Rev. V.17, [http://www.kingbright.com/attachments/file/psearch/000/00/00/KTIR0711S\(Ver.17\).pdf](http://www.kingbright.com/attachments/file/psearch/000/00/00/KTIR0711S(Ver.17).pdf).
- [8] H. Kudela. Aproksymacja funkcji metodą najmniejszych kwadratów. http://fluid.itcmp.pwr.wroc.pl/~znmp/dydaktyka/metnum/najmniejsze_kwadraty.pdf.
- [9] W. Lipieta. Badania porównawcze algorytmów śledzenia trasy dla robotów klasy linefollower. Praca magisterska, Politechnika Wrocławska, 2017.
- [10] R. Merry, M. van de Molengraft, M. Steinbuch. Velocity and acceleration estimation for optical incremental encoders. <https://www.sciencedirect.com/science/article/pii/S0957415809001214>, Luty 2009.
- [11] Microchip. *RN4678*. <http://ww1.microchip.com/downloads/en/DeviceDoc/50002519A.pdf>.
- [12] R. Petrella, M. Tursini, L. Peretti, M. Zigliotto. Speed measurement algorithms for low-resolution incremental encoder equipped drives: a comparative analysis. <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4510607>, Wrze. 2007.

- [13] Pololu Robotics and Electronics. *10:1 Micro Metal Gearmotor HPCB 6V*. <https://www.pololu.com/product/3061>.
- [14] Renishaw. *AM4096 Datasheet*, Mar. 2010. Issue 2, <https://www.rls.si/en/fileuploader/download/download/?d=0&file=custom/upload/File-1406277392.pdf>.
- [15] W. Salejda, R. Poprawski. Podstawy analizy niepewności pomiarowych w studenckim laboratorium podstaw fizyki. http://lpf.wppt.pwr.edu.pl/informacje/an_n.pdf, Mar. 2009.
- [16] STMicroelectronics. *L3GD20 Datasheet*, Luty 2013. Rev. 2, <https://www.st.com/resource/en/datasheet/l3gd20.pdf>.
- [17] STMicroelectronics. *LSM303C Datasheet*, Czerw. 2014. Rev. 2, <https://www.st.com/resource/en/datasheet/lsm303c.pdf>.
- [18] STMicroelectronics. *STM32F767ZI Reference Manual*, Mar. 2018. Rev. 4, https://www.st.com/resource/en/reference_manual/dm00224583.pdf.
- [19] T. P. Zieliński, P. Korohoda, R. Rumian. *Cyfrowe przetwarzanie sygnałów w telekomunikacji*. Wydawnictwo Naukowe PWN SA, 2014.