

Politechnika Warszawska

WYDZIAŁ ELEKTRONIKI
I TECHNIK INFORMACYJNYCH



Instytut Automatyki i Informatyki Stosowanej

Praca dyplomowa inżynierska

na kierunku Automatyka i Robotyka

Generacja trajektorii o trapezoidalnym profilu prędkości w systemie VelmOS

Karolina Borkowska

Numer albumu 270800

promotor
dr inż. Tomasz Winiarski

WARSZAWA 2018

Streszczenie

Tytuł: Generacja trajektorii w systemie IRPOS

Celem pracy inżynierskiej była modyfikacja systemu IRPOS, umożliwiająca sprawdzenie, czy w trakcie osiągania przez robota IRp6 punktów zadanej drogi nie zostaną naruszone jego ograniczenia fizyczne i dynamiczne. A priori przeanalizowana zostaje trajektoria, którą poruszać się ma robot, by osiągnąć zadane położenie. Zmodernizowany został komponent OROCOS, odpowiedzialny za odbieranie i wstępne sprawdzenie zadania robota. Dodatkowo nowa funkcjonalność wstrzymuje dalsze przetwarzanie rozkazu i związany z nim ruch. Jest to podejście zgodne z wcześniej zaimplementowaną koncepcją reakcji na zadanie punktu końcowego niemożliwego do osiągnięcia dla danego stawu.

Słowa kluczowe: robot IRp6, manipulator robotyczny, system IRPOS, planowanie trajektorii manipulatora, OROCOS, Open Robot Control Software.

Abstract

Title: Generation of robot trajectory in IRPOS system

The aim of this thesis was modifying IRPOS system, that enables checking whether, during IRp6 robot's movement in the direction of setpoints, the physical and dynamic limitations of the robot are not breached. A priori a possible trajectory is analysed. OROCOS component, responsible for receiving and initial movement task test, was modernised. Moreover, the new functionality blocks further setpoint message processing, thus halting robot's movement. This approach is in accordance with the previously implemented reaction to a setpoint that is impossible to reach by a given joint.

Keywords: IRp6 robot, manipulator robot, IRPOS system, manipulator trajectory planning, OROCOS, Open Robot Control Software .



Politechnika Warszawska

załącznik do zarządzenia nr 28/2016 r.
Rektora PW

„załącznik nr 3 do zarządzenia nr 24/2016 Rektora PW

.....
miejscowość i data

.....
imię i nazwisko studenta

.....
numer albumu

.....
kierunek studiów

OŚWIADCZENIE

Świadomy/-a odpowiedzialności karnej za składanie fałszywych zeznań oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie, pod opieką kierującego pracą dyplomową.

Jednocześnie oświadczam, że:

- niniejsza praca dyplomowa nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 roku o prawie autorskim i prawach pokrewnych (Dz.U. z 2006 r. Nr 90, poz. 631 z późn. zm.) oraz dóbr osobistych chronionych prawem cywilnym,
- niniejsza praca dyplomowa nie zawiera danych i informacji, które uzyskałem/-am w sposób niedozwolony,
- niniejsza praca dyplomowa nie była wcześniej podstawą żadnej innej urzędowej procedury związanej z nadawaniem dyplomów lub tytułów zawodowych,
- wszystkie informacje umieszczone w niniejszej pracy, uzyskane ze źródeł pisanych i elektronicznych, zostały udokumentowane w wykazie literatury odpowiednimi odnośnikami,
- znam regulacje prawne Politechniki Warszawskiej w sprawie zarządzania prawami autorskimi i prawami pokrewnymi, prawami własności przemysłowej oraz zasadami komercjalizacji.

Oświadczam, że treść pracy dyplomowej w wersji drukowanej, treść pracy dyplomowej zawartej na nośniku elektronicznym (płyce kompaktowej) oraz treść pracy dyplomowej w module APD systemu USOS są identyczne.

.....
czytelny podpis studenta”

Spis treści

1	Wstęp	3
1.1	Wprowadzenie	3
1.2	Motywacja	3
1.3	Cel pracy	4
1.4	Struktura pracy	4
2	Wykorzystany sprzęt i narzędzia	5
2.1	Robot IRp-6	5
2.2	Robot Velma	5
2.3	Oprogramowanie bazowe	6
2.3.1	ROS	6
2.3.2	Open Robot Control Software	8
2.3.3	Connman	8
2.3.4	Rviz	8
2.3.5	Gazebo	8
2.4	System IrpOS	8
2.5	System VelmOS	8
2.5.1	rqt_agent	8
2.5.2	show_collisions	8
2.5.3	show_joints	8
2.6	Algorytmy	8

Rozdział 1

Wstęp

1.1 Wprowadzenie

Mechanizmy imitujące ludzi i zwierzęta wydają się być częścią wspólnej kulturowej świadomości od początku istnienia cywilizacji. Mitologie Egipcjan, Greków, Hindusów czy Izraelitów wspominają o urządzeniach, które współczesny czytelnik mógłby nazwać robotami. Największe umysły w historii aspirowały do stworzenia pierwszego autonomicznego pracownika. Wśród pierwszych konstruktorów można znaleźć Leonardo da Vinci’ego, czy Archytas’a z Tarentu. To, co dla naszych poprzedników pozostawało jedynie fikcją, stało się nieodzowną częścią przemysłu w XX wieku. Już pierwsze roboty firmy Unimation cechowały się dokładnością znacznie przewyższająca możliwości ludzkie, dodatkowo zapewniając szybszą produkcję oraz większe bezpieczeństwo pracowników.

Jednym z najczęściej spotkanych typów robotów przemysłowych jest manipulator. W swej budowie przypominają one górne kończyny ludzkie. Zakończone są chwytakami, które choć z zasady mają być odwzorowaniem ludzkiej dłoni, są raczej jej dużym uproszczeniem lub wręcz modyfikacją wprowadzoną do wykonania konkretnego zadania. Człowiek intuicyjnie wybiera rozwiązanie przypominające wytwory natury, przez co manipulatory znajdują szerokie zastosowanie w najróżniejszych gałęziach przemysłu. Sprzęt taki jak kamery, taśmy produkcyjne lub tor jezdny, to podstawowe rozszerzenia, jakie dodawane są do systemów robotycznych, w celu automatyzacji kolejnych etapów produkcji. Od fabryk samochodowych, po te skupiające się na wyrobach spożywczych, manipulatory są integralną częścią nowoczesnej cywilizacji.

1.2 Motywacja

System VelmOS zapewnia prosty interfejs do pracy z robotem Velma (oraz jego symulacją) znajdującym się w laboratorium P109 Wydziału Elektroniki i Technik

Informacyjnych. Wykorzystywany on jest do pisania prac naukowych oraz prowadzenia zajęć dydaktycznych. Oprogramowanie to jest wciąż rozwijane, a nowe zapotrzebowania są definiowane w trakcie tego procesu. Obecnie zaimplementowane są dwa rodzaje generatorów trajektorii: spline’owy oraz kartezjański. Stworzone są dla nich oddzielne stany subsystemu `velma_core_cs`. Jednakże żaden z nich nie zapewnia odpowiedniego stopnia kontroli profilu prędkościowego w każdym ze stawów. Nowa funkcjonalność, oparta na generatorze trajektorii o innych właściwościach, mogłaby znaleźć zastosowanie przy różnorodnych badaniach naukowych oraz analizowaniu samego robota.

1.3 Cel pracy

Celem niniejszej pracy było stworzenie dodatkowej funkcjonalności systemu VelmaOS, która pozwoliłaby na większą kontrolę osiąganą prędkości w kolejnych stawach robota. Pracę rozpoczęto od alegorycznej modyfikacji systemu IrpOS, gdyż jest on prostszą wersją oprogramowania robota Velma. Zdecydowano zaimplementować dodatkowy generator trajektorii, istniejący w systemie równolegle z już istniejącymi generatorami. Jego funkcjonalność rozszerzano poprzez definiowanie trybów i podtrybów działania. Nowy generator produkuje trajektorię o profilu trapezoidalnym. System dostosowano do pracy w trybie „trapezoidalnym” nowymi funkcjami interfejsu i zmianami wprowadzonymi do plików konfiguracyjnych. Na sam generator nałożono restrykcje, które często są charakterystyczne dla tego typu ruchu. Naruszenie którejkolwiek z nich powoduje specyficzną odpowiedź dla użytkownika. W pracy ograniczono się do przeprowadzania badań za pomocą symulacji.

1.4 Struktura pracy

Praca składa się z sześciu rozdziałów. Rozdział 2 przedstawia sprzęt i narzędzia wykorzystywane do stworzenia pracy. Rozdział 3 prezentuje projekt rozwiązania wybranego problemu. Rozdział 4 opisuje jak zrealizowano zaproponowane rozwiązanie. Rozdział 5 skupia się na sposobie i wynikach badania poprawności działania nowej funkcjonalności. Rozdział 6 podsumowuje wyniki pracy.

Rozdział 2

Wykorzystany sprzęt i narzędzia

2.1 Robot IRp-6

Laboratorium 012 Wydziału Elektroniki i Technik Informacyjnych wyposażone jest w dwa roboty typu IRp-6. Bazują one na manipulatorach IRb-6, o pięciu stopniach swobody[1], które odznaczyły się w historii robotyki jako pierwsze elektryczne roboty przemysłowe sterowane za pomocą mikroprocesorów[2]. Oba manipulatory Politechniki rozbudowano o kiście pozwalające na obrót nadgarstka, a jeden z nich dodatkowo może poruszać się wzdłuż toru jezdnego. Ostatecznie manipulatory *Postument* i *Track* posiadają kolejno sześć oraz siedem stopni swobody[3]. Zastosowanie czujników położenia stawów pozwala na precyzyjny ruch, szczególnie ważny przy zadaniach przemysłowych i badawczych.

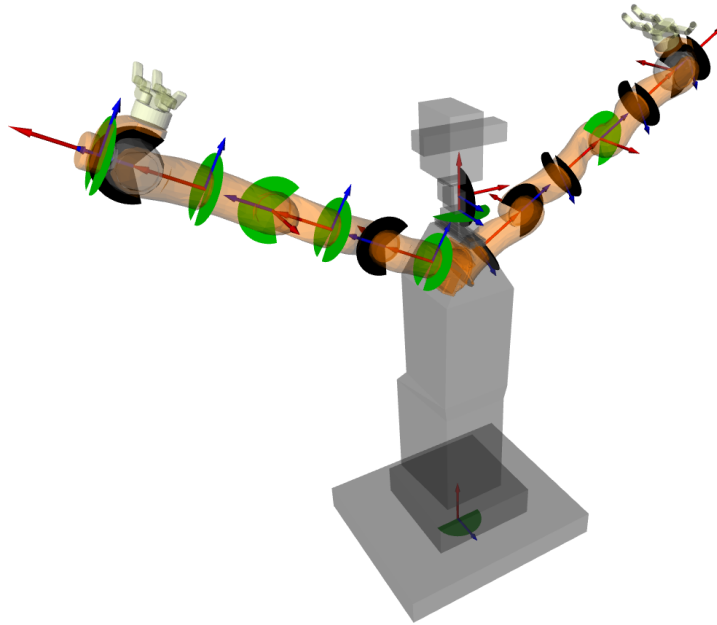
2.2 Robot Velma

Robot Velma składa się z czterech części:

1. obrotowego tułowia;
2. ramion KUKA LWR;
3. chwytaków BarrettHand;
4. szyi, na której zamieszczona jest głowa z kamerą Kinect.

Różnią się one między sobą stopni swobody oraz sposobem sterowania. Elementy nr 1 i 2 steruje się impedancyjnie, pozostałe pozycyjnie[4]. Pełną strukturę sprzętu pokazuje rysunek 2.1.

Tak jak w przypadku robotów Irp-6, szeroka gama czujników pozwala na dokładne sterowanie. Zaś sterowanie impedancyjne, a konkretnie nadawanie pożądanej sztywności stawom, daje możliwość manipulacji dynamicznych relacji robota ze środowiskiem. Dodatkowo system Velmy wywołuje wirtualne siły odpychające



Rysunek 2.1: Robot Velma (źródło: [4])

od siebie staw, tak by zapobiec ich kolizji. Zarówno obniżanie sztywności w stawach, jak i wprowadzanie wirtualnych sił zmniejsza prawdopodobieństwo idealnego osiągnięcia zadanej pozycji.

2.3 Oprogramowanie bazowe

Oprogramowanie, które steruje robotami Irp-6 i Velma opiera się na wolnodostępnych platformach. Wprowadzone modyfikacje musiały zostać stworzone w tej samej konwencji i technologii.

2.3.1 ROS

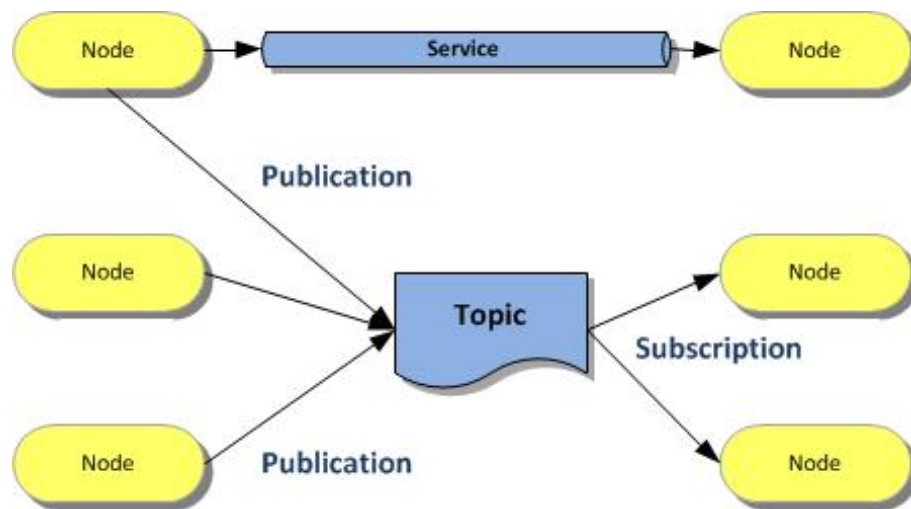
ROS (The Robot Operating System) to otwarta platforma programistyczna, która przede wszystkim definiuje sposoby komunikacji między różnymi fragmentami systemu[5]. Poza tym implementuje narzędzia przydatne nie tylko do czystej pracy systemu, ale i jego analizy.

ROS rozdziela elementy na węzły. Założeniem systemu jest dzielenie fragmentów oprogramowania na podstawie funkcjonalności. Przykładowo analizę położenia robota można rozdzielić na kolejne węzły: zbierający informację od enkoderów/kamer, konwertujący odczyty do bardziej czytelnej formy, określający położenie na podstawie dostosowanych danych. Widać tu potrzebę przesyłu danych pomiędzy wymienionymi węzłami. Istnieją dwa podstawowe sposoby komunikacji węzłów w systemie opartym na tej platformie:

- za pomocą tematów - jeden z węzłów pisze informacje na temacie (publisher), inne mogą je z niego odczytywać (subscriber);

- komunikacja jeden-do-jednego - jeden z węzłów zapewnia serwis, odpowiadający na zapytanie klienta.

Nie jest wykluczone by węzeł był jednocześnie odbiorcą i dostarczycielem danych. Tak samo możliwe jest by komunikacja z węzłem przebiegała na oba sposoby. Sposoby komunikacji przedstawiono jest na rysunku 2.2.



Rysunek 2.2: Podstawowa komunikacja w systemie ROS (źródło: [6])

Dodatkowo ROS wyposażony jest w bibliotekę `actionlib`. Jest to interfejs dla zadań, które można wywłaszczyć[7]. Jest to szczególnie przydatne gdy wykonanie czynności jest długotrwałe i może zaburzyć działanie systemu. Wiadomości przekazywane pomiędzy klientem i serwisem w plikach deklaracji podzielone są na trzy części: `goal`, informacje zwrotne oraz wynik zadania. Serwer może w danym momencie mieć tylko jeden aktywny `goal`, o którego statusie wysyłane są informacje zwrotne (np. w każdej iteracji obliczającej ustawienia chwilowe). Biblioteka zapewnia wygodny sposób na odwoływanie się do pól struktury wiadomości akcji oraz do operowania statusem celu (ustawianie go na przyjęty, odrzucony itp.). Ten sposób komunikacji jest wykorzystywany do łączenia interfejsu użytkownika systemu IrpOS lub VelmOS z komponentami akcji, a przez to dedykowanymi im generatorami.

2.3.2 Open Robot Control Software

2.3.3 Connman

2.3.4 Rviz

2.3.5 Gazebo

2.4 System IrpOS

2.5 System VelmOS

2.5.1 rqt_agent

2.5.2 show_collisions

2.5.3 show_joints

2.6 Algorytmy

Bibliografia

- [1] Przemysłowy Instytut Automatyki i Pomiarów MERA-PIAP. *Dokumentacja techniczno-ruchowa robotów IRb-6 i IRb-60*, 1978.
- [2] ABB Robotics Historical milestones. <https://new.abb.com/products/robotics/home/about-us/historical-milestones>.
- [3] Wiki systemu IrpOS, 2016. https://github.com/RCPRG-ros-pkg/irp6_robot/wiki.
- [4] Documentation of control system of WUT Velma Robot, 2017. https://rcprg-ros-pkg.github.io/velma_docs/2017/10/13/01_introduction_02_velma_robot/.
- [5] Oficjalna strona systemu ROS. <http://ros.org>.
- [6] What is ROS? <https://www.generationrobots.com/blog/en/ros-robot-operating-system-2/>.
- [7] Dokumentacja biblioteki actionlib. <http://wiki.ros.org/actionlib>.