

Instytut Informatyki, Automatyki i Robotyki

Seria: PREPRINTY nr 92/2011

**Aproksymacja algorytmów planowania
ruchu robotów**

(rozprawa doktorska)

Joanna Karpińska

Promotor: prof. dr hab. inż. Krzysztof Tchoń

Słowa kluczowe:

- manipulator redundantny;
- odwrotne zadanie kinematyki;
- algorytm jakobinowy;
- aproksymacja;
- metoda homotopii;
- zadanie reprodukcji trajektorii.

Wrocław, 2011-11-28

POLITECHNIKA WROCŁAWSKA
WYDZIAŁ ELEKTRONIKI

ROZPRAWA DOKTORSKA

Aproksymacja algorytmów planowania ruchu
robotów

Joanna Karpińska

Promotor:
prof. dr hab. inż. Krzysztof Tchoń

14 11 2011

Wyrażam głęboką wdzięczność Panu Profesorowi dr hab. inż. Krzysztofowi Tchoniowi za nieocenioną pomoc, wsparcie, cierpliwość oraz poświęcony mi czas.

Dziękuję Rodzicom za wiarę we mnie i pomoc w realizacji zamierzonych celów.

Podziękowania kieruję także do Adama za cenne uwagi, optymizm i uśmiech.

Spis treści

Spis ważniejszych oznaczeń	3
1 Wstęp	5
2 Aproksymacja jacobianowych algorytmów kinematyki odwrotnej	11
2.1 Preliminaria	11
2.2 Metoda wariacyjna	14
2.2.1 Funkcjonał błędu Robertsa i Maciejewskiego	14
2.2.2 Interpretacja	16
2.2.3 Funkcjonał błędu I	18
2.2.4 Funkcjonał błędu II	22
2.3 Metoda geometryczna	23
2.4 Metody rozwiązania	25
2.4.1 Metoda pośrednia — rachunek wariacyjny	25
2.4.2 Metoda bezpośrednia — metoda Ritza	27
3 Zastosowanie metody homotopii do zadania reprodukcji trajektorii układów robotycznych	29
4 Przykłady numeryczne — aproksymacja algorytmów kinematyki odwrotnej	33
4.1 Manipulator POLYCRANK	33
4.2 Metoda wariacyjna — Funkcjonał błędu I...	35
4.2.1 Manipulator 3R bez konfiguracji osobliwych	35
4.2.2 Manipulator TTR	38
4.2.3 Manipulator POLYCRANK	42
4.3 Metoda wariacyjna — Funkcjonał błędu I...	49
4.3.1 Podmanipulator stanfordzki	49
4.3.2 Manipulator POLYCRANK	53
4.4 Metoda wariacyjna — Funkcjonał błędu II	59
4.4.1 Manipulator POLYCRANK	59
4.4.2 Monocykl	64
4.5 Metoda geometryczna	68
4.5.1 Manipulator POLYCRANK	68
4.6 Porównanie metody geometrycznej i wariacyjnej	73
4.7 Aspekty obliczeniowe	81

Spis ważniejszych oznaczeń

$x \in \mathbb{R}^n$	wektor zmiennych stanu/przegubowych
$y \in \mathbb{R}^m$	wektor zmiennych zadaniowych
$u \in \mathbb{R}^r$	wektor sterowań
$u(\cdot) \in L_T^2[0, T]$	funkcja sterująca, całkowalna z kwadratem na odcinku $[0, T]$
s	stopień redundancji, $s = n - m$
$T(x)$	wektor translacji
$R(x)$	macierz rotacji
$k(x)$	reprezentacja kinematyki we współrzędnych
$\mathcal{K}(x)$	kinematyka w $SE(3)$
$h(x)$	funkcja rozszerzająca
e	błąd w przestrzeni zadaniowej
$J(x)$	jakobian analityczny
$J_{RT}(x)$	jakobian geometryczny
$J^{P\#}(x)$	pseudoodwrotność jakobianu
$J^{E\#}(x)$	odwrotność typu jakobianu rozszerzonego
I_m	macierz jednostkowa wymiaru $m \times m$
$\mathcal{D}_P$	dystrybucja stowarzyszona z algorytmem jakobianu pseudoodwrotnego
$\mathcal{D}_E$	dystrybucja stowarzyszona z algorytmem jakobianu rozszerzonego
$\mathcal{D}_P^\perp$	kodystrybucja stowarzyszona z algorytmem jakobianu pseudoodwrotnego
$\mathcal{D}_E^\perp$	kodystrybucja stowarzyszona z algorytmem jakobianu rozszerzonego
$\ A\ $	norma operatorowa operatora A
$\ M\ _F$	norma Frobeniusa macierzy M
$\mathbb{Q}$	obszar aproksymacji zawarty w zbiorze konfiguracji regularnych
Ω	forma różniczkowa
$[X, Y]$	nawias Liego pól wektorowych
$K(x)$	macierz, której kolumny generują jądro jakobianu
L	funkcja Lagrange'a
$G(x)$	macierz pól wektorowych układu sterowania
$f(x)$	dryf układu sterowania
$A(t), B(t), C(t)$	macierze przybliżenia liniowego układu sterowania
$\Phi(t, s)$	macierz fundamentalna
$J_{x_0, t}$	jakobian stowarzyszony z afinicznym układem sterowania
$\theta \in \mathbb{R}$	parametr homotopii
$H(t, \theta)$	funkcja homotopii

Rozdział 1

Wstęp

Termin *robot* (czes. *robota* - praca) oznacza cybernetyczną maszynę przeznaczoną do realizacji niektórych czynności energetyczno-ruchowych, sensualnych i intelektualnych człowieka. Termin powstał już w 1920r. i został wprowadzony przez czeskiego fantastyka Karelą Čapka w jego sztuce „R.U.R.” („Rossumovi Univerzální Roboti”). Z kolei termin *robotyka* wprowadził w 1942r. amerykański pisarz rosyjskiego pochodzenia Isaac Asimov (Айзек Азимов) w opowiadaniu „Runaround”. Równolegle z wprowadzaniem pojęć zaczęły także powstawać pierwowzory robotów. Pierwsze konstrukcje robotyczne datuje się na okres międzywojenny, kiedy to powstał na przykład „pies elektroniczny” zbudowany w 1929r. Po II wojnie światowej postęp techniki przyczynił się także do rozwoju dyscypliny robotyka. Pierwsze modele robotów przeznaczonych do pracy w przemyśle zostały zbudowane w Stanach Zjednoczonych na początku lat sześćdziesiątych. W Europie roboty pojawiły się niedługo później. Pierwszy europejski robot został zastosowany w 1971r. w Szwecji, a pierwszy robot w Polsce wspomógł pracę pracowników Olkuskiej Fabryki Naczyni Emaliowanych w 1976r. Obecnie robotyka oprócz tematów związanych z największą grupą robotów – manipulacyjnych robotów przemysłowych, zajmuje się także zagadnieniami robotów mobilnych i innych układów mechanicznych. Ostatnio, robotyka wkracza także w obszary psychologiczno-socjologiczne pracując nad robotami społecznymi. Zadania stawiane robotom mają różnoraki charakter. Od prostych i odwrotnych zadań kinematyki, poprzez takie zadania jak, planowanie ruchu czy śledzenie trajektorii, aż po skomplikowane zadania jak na przykład wyrażanie emocji.

Proste i odwrotne zadanie kinematyki należą do fundamentalnych zadań w robotyce [4, 29, 48]. Proste zadanie kinematyki dla manipulatorów polega na zdefiniowaniu pozycji (np. położenia i orientacji) końca efektora jako funkcji zmiennych przegubowych. Natomiast, celem odwrotnego zadania kinematyki jest wyznaczenie wartości zmiennych przegubowych dla określonej pozycji końca efektora. Odwrotne zadanie kinematyki jest szczególnym przypadkiem zadania planowania ruchu. Dla manipulatorów redundantnych, czyli dla takich, w których wymiar przestrzeni konfiguracyjnej jest większy od wymiaru przestrzeni zadaniowej, odwrotne zadanie kinematyki ma nieskończenie wiele rozwiązań. Aby otrzymać jednoznaczne rozwiązanie algorytmu kinematyki odwrotnej, nakłada się na rozwiązanie dodatkowe ograniczenia, jak na przykład minimalizacja prędkości przegubowych w trakcie ruchu, czy inne jak unikanie przeszkód, ograniczeń w przegubach, bądź konfiguracji osobliwych. Rozważania na temat kinematyki robotów redundantnych przedstawione są w [12]. W większości przypadków odwrotne zadanie kinematyki dla manipulatorów redundantnych jest rozwiązywane numerycznie, za pomocą jakobianowych algorytmów kinematyki odwrotnej, takich jak algorytm jakobianu pseudoodwrotnego, bądź algorytm jakobianu rozszerzonego. Z definicji, pierwszy z nich cechuje minimalizacja prędkości

przegubowych w czasie trwania ruchu oraz szybka zbieżność. Drugi z nich, konstruowany jest poprzez dodanie odpowiedniej ilości wierszy będących różniczką funkcji rozszerzających, w taki sposób, aby wynikowy jacobian był macierzą kwadratową [7]. Odpowiedni dobór funkcji rozszerzających pozwala kształtować dynamikę algorytmu. Rozwiązanie algorytmu typu jacobianu rozszerzonego zależy od obranej funkcji rozszerzającej. Dlatego, algorytm ten poza rozwiązaniem odwrotnego zadania kinematyki może jednocześnie minimalizować pewną funkcję kryterialną [25]. Z tego samego powodu, algorytm jacobianu rozszerzonego posiada bardzo istotną cechę jaką jest powtarzalność [43]. Gwarantuje ona, że algorytm będzie generował powtarzalne rozwiązania. Innymi słowy, zamkniętym ścieżkom w przestrzeni zadaniowej odpowiadać będą zamknięte ścieżki w przestrzeni konfiguracyjnej. Własność ta jest niezwykle przydatna podczas planowania ruchów cyklicznych. Ruchy takie są wymagane od robotów przemysłowych pracujących na liniach produkcyjnych, czy montażowych. Powszechnie wiadomo, że algorytm jacobianu pseudoodwrotnego nie posiada cechy powtarzalności [26]. Alternatywą dla algorytmów jacobianowych jest algorytm oparty na zastosowaniu mnożników Lagrange’a [10]. W pracy [24] można znaleźć porównanie algorytmu jacobianu rozszerzonego i podejścia korzystającego z mnożników Lagrange’a. Problem osobliwości algorytmu jacobianu rozszerzonego został poruszony w [25], a następnie zdefiniowany na nowo w [24].

Przy wykorzystaniu własności algorytmu jacobianu rozszerzonego jest możliwe zbudowanie powtarzalnego algorytmu kinematyki odwrotnej, który będzie przypominał w działaniu inny algorytm, w szczególności algorytm jacobianu pseudoodwrotnego. W tym celu, definiuje się miarę, która określa podobieństwo dwóch algorytmów. Następnie, rozwiązuje się zadanie aproksymacji polegające na wyznaczeniu funkcji rozszerzającej minimalizującej kryterium odległości. Po raz pierwszy takie zadanie zostało sformułowane dla manipulatorów stacjonarnych przez Robertsa i Maciejewskiego w serii artykułów [38–40]. Wspomniana literatura jest dedykowana problemowi optymalnej aproksymacji algorytmu jacobianu pseudoodwrotnego przez powtarzalny algorytm jacobianowy. Autorzy zdefiniowali zadanie aproksymacji jako zadanie optymalizacji w przestrzeni funkcyjnej, które rozwiązywali korzystając z metod rachunku wariacyjnego. Aby otrzymać optymalną funkcję rozszerzającą, należy rozwiązać układ nieliniowych cząstkowych równań różniczkowych, wynikających z równania Eulera–Lagrange’a. Rozwiązanie tak określonego układu równań jest zadaniem bardzo trudnym, nawet dla manipulatora o kilku stopniach swobody. W przypadku bardziej skomplikowanych manipulatorów rozwiązanie zadania optymalizacji staje się wręcz niemożliwe, biorąc pod uwagę nie tylko metody analityczne, ale także dostępne metody numeryczne. W pracy [53] zaproponowano modyfikację metody Robertsa i Maciejewskiego, której rozwiązanie za pomocą metody Eulera–Lagrange’a sprowadza się do wyznaczenia funkcji rozszerzającej poprzez rozwiązanie układu liniowych, eliptycznych równań różniczkowych cząstkowych. Sformułowanie zadania aproksymacji dla kinematyki w $SE(3)$ zawarte jest w [22]. Praca [54] przedstawia nowe podejście do problemu, zaczerpnięte z geometrii różniczkowej, w której zadanie aproksymacji sprowadza się do zadania aproksymacji kodystrybucji. Istotą metody jest zdefiniowanie całkowalnej kodystrybucji pokrywającej się z daną, niecałkowalną kodystrybucją na pewnym obszarze w przestrzeni konfiguracyjnej manipulatora. Inspiracją do stworzenia podstaw teoretycznych tej metody jest teoria sterowania, w której rozważany jest problem przybliżonej linearyzacji poprzez sprzężenie zwrotne [46]. Przykład zadania aproksymacji algorytmu jacobianu pseudoodwrotnego algorytmem jacobianu rozszerzonego zdefiniowanego zarówno za pomocą podejścia wariacyjnego jak i geometrii różniczkowej zawarto w pracy [56].

Niniejsza praca dotyczy podstaw robotyki i odnosi się do matematycznych podstaw algorytmów stosowanych w robotyce. W rozprawie zostaną przedstawione metody aprok-

symacji algorytmu jacobianu pseudoodrotnego jacobianem rozszerzonym korzystające zarówno z metod rachunku wariacyjnego jak i geometrii różniczkowej. Rozpocznijmy od wyprowadzenia algorytmów jacobianowych metodą homotopii definiującą algorytm jako układ dynamiczny zależny od funkcji błędu określonego w przestrzeni zadaniowej, dla którego trajektoria zbiega się z rozwiązaniem odwrotnego zadania kinematyki [14, 52]. W rozprawie będziemy rozpatrywać zadania aproksymacji korzystające z metod rachunku wariacyjnego lub z geometrii różniczkowej. Należy pamiętać, że wszystkie metody operują na podzbiorze konfiguracyjnym wolnym od osobliwości. „Funkcjonał błędu I” metody wariacyjnej swoje podstawy czerpie z koncepcji Robertsa i Maciejewskiego. Zakłada ona jednak trochę inne podejście do problemu. Polega ono na osadzeniu odwrotności jacobianów w większej przestrzeni. Wówczas, zadanie aproksymacji jest formułowane jako zadanie optymalizacji w przestrzeni funkcyjnej, które następnie rozwiązujemy przy pomocy rachunku wariacyjnego. W efekcie, otrzymujemy układ równań różniczkowych cząstkowych dla funkcji rozszerzającej. Dla metody opracowanej przez Robertsa i Maciejewskiego układ równań tworzą nieliniowe równania różniczkowe cząstkowe. W przypadku nowego podejścia układ ten zawiera liniowe, eliptyczne równania różniczkowe cząstkowe, którego rozwiązaniem mogą być funkcje harmoniczne [6, 19]. Warto zwrócić uwagę na fakt, że jak dotąd, funkcje harmoniczne pojawiły się w dziedzinie robotyki w dwóch innych kontekstach: jako funkcje potencjału w zadaniu planowania ruchu manipulatora [23] i jako rozwiązanie zadania optymalizacji dystorsji kinematycznej manipulatora [32]. Celem „Funkcjonału błędu II” podejścia wariacyjnego i metody geometrycznej jest aproksymacja kodystrybucji stowarzyszonych z odwrotnościami jacobianów pseudoodrotnego i rozszerzonego. W przypadku małej redundancji rozpatrywanej kinematyki, wygodnie jest dystrybucję zastąpić kodystrybucją rozpiętą przez 1-formy różniczkowe anihilujące stowarzyszoną dystrybucję. W przypadku jacobianu pseudoodrotnego kodystrybucja ta jest rozpięta przez 1-formy anihilujące jacobian transponowany. Natomiast, dla algorytmu jacobianu rozszerzonego stowarzyszona kodystrybucja jest rozpinana przez odpowiednie różniczki funkcji rozszerzającej kinematykę. Zadanie aproksymacji polega na znalezieniu dystrybucji związanej z odwrotnością jacobianu rozszerzonego, która jest w pewnym sensie bliska danej dystrybucji stowarzyszonej z jacobianem pseudoodrotnym. Zgodnie z podejściem geometrycznym, obszar nieosobliwy przestrzeni konfiguracyjnej jest przekształcony w foliację, której liście, o wymiarze równym stopniowi redundancji, są homotopijne do liścia odniesienia. Homotopia ta definiuje pewne pole wektorowe. Kodystrybucja stowarzyszona z jacobianem rozszerzonym powinna pokrywać się z kodystrybucją stowarzyszoną z jacobianem pseudoodrotnym określoną na liściach foliacji i wzdłuż pola wektorowego określonego przez homotopię. Warunek ten sprowadza się do układu równań różniczkowych cząstkowych dla funkcji rozszerzających, które rozwiązuje się przy pomocy metody charakterystyk. Ostatecznie, funkcja rozszerzająca jest obliczana numerycznie poprzez całkowanie wstecz równania charakterystycznego. W przeciwieństwie do „Funkcjonału błędu I”, gdzie funkcjonałem jest odległość pomiędzy jacobianami, w „Funkcjonałe błędu II” konstruowany jest funkcjonal błędu aproksymacji będący miarą odległości pomiędzy dwiema dystrybucjami/kodystrybucjami na podzbiorze nieosobliwym przestrzeni konfiguracyjnej. Warunek optymalności wyrażony poprzez równanie Eulera–Lagrange’a jest równoważny liniowemu, eliptycznemu równaniu różniczkowemu cząstkowemu. Ponieważ liczba zmiennych niezależnych tego równania jest równa liczbie stopni swobody np. manipulatora, rozwiązanie tak sformułowanego zadania może być niemożliwe przy użyciu obecnie dostępnych narzędzi numerycznych. Z tego powodu, do rozwiązania zadania aproksymacji dla metod wariacyjnych posłużono się metodą Ritza [17].

Metoda homotopii (zwana także metodą kontynuacji), która leży u podstaw algo-

rytmów jacobianowych znalazła ciekawe zastosowanie do syntezy algorytmów reprodukcji trajektorii układu. Metoda homotopii jest jednym z narzędzi często stosowanych we współczesnej matematyce. W ogólności, metoda homotopii opiera się na zaleceniu, że *jeśli nie potrafisz rozwiązać danego problemu, staraj się go uogólnić*. W ten sposób, oryginalny problem jest niejako zanurzany w rodzinie podobnych problemów parametryzowanych pewną zmienną. Mając rozwiązanie problemu dla zadanej wartości parametru, przez przejście z parametrem do granicy (w szczególności do $+\infty$) otrzymujemy rozwiązanie problemu oryginalnego. Metoda homotopii znalazła zastosowanie przy rozwiązywaniu problemów optymalizacyjnych [20, 60, 61], problemów równoważności [52] i zagadnień odwrotnych [2, 37, 51, 52]. Na jej podstawie powstały także efektywne algorytmy do obliczeń numerycznych [1]. Pierwsze zastosowanie metody homotopii w robotyce przedstawił Sussmann w pracy [50]. Wykazał on, że przy pomocy metody homotopii można rozwiązać odwrotne zadanie kinematyki dla układów nieholonomicznych spełniających warunek Strong Bracket Generating Condition [49]. Równolegle Wen i współpracownicy prowadzili prace nad wykorzystaniem tej metody do planowania ruchu manipulatorów redundantnych [41], a także platform mobilnych [15, 16]. W późniejszym czasie pojawiły się prace dotyczące planowania ruchu samochodu kinematycznego [14] i ciał toczących się [3, 11]. Dogłębne rozważania na temat zastosowania metody homotopii do planowania ruchu robotów pojawiły się w [13, 14]. Korzystając z idei homotopii rozwinęła się także metoda endogenicznej przestrzeni konfiguracyjnej [34, 35, 57], która umożliwia zastosowanie pojęć teorii manipulatorów stacjonarnych do robotów mobilnych.

W niniejszej pracy zaadaptujemy metodę homotopii do rozwiązania zadania planowania, a także reprodukcji trajektorii układów robotycznych. Zadanie planowania trajektorii możemy sformułować następująco: mając dany afiniczny układ sterowania, wyznaczyć funkcję sterującą, przy pomocy której układ będzie poruszał się wzdłuż zadanej trajektorii [27]. Rozwiązanie zadania planowania trajektorii wymaga pewnego rodzaju inwersji układu sterowania, a dokładniej, inwersji odwzorowania wejściowo-wyjściowego dla określonego stanu początkowego. W przypadku, gdy to odwzorowanie jest iniektywne (różnowartościowe), system taki nazywamy lewostronnie odwracalnym. Natomiast, gdy odwzorowanie jest suriektywne nazywa się go prawostronnie odwracalnym. Ta ostatnia cecha nazywana jest także reprodukowalnością funkcjonalną [9]. W literaturze istnieje wiele pozycji koncentrujących się na inwersji układów sterowania, zarówno liniowych [44] jak i nieliniowych [45]. Należy podkreślić, że jeśli układ jest różniczkowo płaski [33] i znamy trajektorię wyjścia, to problem planowania trajektorii jest łatwo rozwiązywalny. Obszerny przegląd teorii i algorytmów odwracania układów sterowania zawiera praca [36].

Zakładając funkcjonalną reprodukowalność, w rozprawie zaproponowane zostaną algorytmy funkcjonalnej reprodukcji stanu i wyjścia afinicznego układu sterowania. Algorytmy te polegają na rozwiązaniu pewnego równania różniczkowego cząstkowego drugiego rzędu, w wyniku czego otrzymujemy jednoparametrową rodzinę sterowań. Następnie, przy parametrze dążącym do $+\infty$ otrzymujemy sterowanie będące rozwiązaniem zadania reprodukcji trajektorii. Algorytmy te mają zastosowanie do afinicznych układów sterowania, które w szczególności mogą reprezentować kinematykę nieholonomicznego robota mobilnego lub dynamikę manipulatora bądź robota mobilnego. Ze względu na przejście graniczne, algorytmy mają charakter asymptotyczny. Zastosowanie metody homotopii do rozwiązania zadania reprodukcji trajektorii stanu można znaleźć w [55], natomiast do zadania reprodukcji trajektorii wyjścia w pozycji [21].

Biorąc pod uwagę powyższe rozważania autorka pracy stawia tezę:

Jest możliwe nowe sformułowanie zadania aproksymacji algorytmów jakobianowych i jego rozwiązanie metodami rachunku wariacyjnego dla redundantnych manipulatorów przemysłowych. Dodatkowo, korzystając z metody homotopii, można skonstruować algorytm reprodukcji trajektorii stanu i wyjścia robota.

Praca składa się z 6 rozdziałów. Rozdział pierwszy zawiera wprowadzenie w tematykę pracy. W tej części zawarty jest także przegląd literatury. W rozdziale drugim przedstawiony został problem aproksymacji algorytmów planowania ruchu robotów, w odniesieniu do algorytmu typu jakobianu rozszerzonego aproksymującego algorytm typu jakobianu pseudoodwrotnego i podano dwie metody jego rozwiązania. Pierwsza metoda wykorzystuje metody rachunku wariacyjnego. Zadanie aproksymacji algorytmów pochodzi od Robertsa i Maciejewskiego. Autorzy ci zdefiniowali miarę odległości jakobianu pseudoodwrotnego i odwrotności typu jakobianu rozszerzonego. Wówczas, zadanie aproksymacji sprowadza się do zadania znalezienia funkcji rozszerzającej minimalizującej przyjętą miarę. W rozwinięciu idei wymienionych autorów został opracowany nowy algorytm aproksymacyjny, oparty na naturalnym rozszerzeniu odwrotności jakobianu rozszerzonego jak i jakobianu pseudoodwrotnego do pewnych macierzy kwadratowych. W tym wypadku, zadanie aproksymacji jest rozwiązywane poprzez skonstruowanie miary odległości rozszerzonych macierzy, a następnie jej minimalizacji ze względu na funkcję rozszerzającą. Wyprowadzone zostało także drugie kryterium, którego minimalizacja pozwala wyznaczyć funkcję rozszerzającą. W tym podejściu, funkcjonal błędu aproksymacji jest miarą odległości między dwiema dystrybucjami/kodystrybucjami w pewnym obszarze przestrzeni konfiguracyjnej. To kryterium służy do porównania wyników uzyskanych metodą wariacyjną i metodą geometryczną. Druga z tych metod korzysta z geometrii różniczkowej. Zadanie aproksymacji algorytmów zostało sformułowane jako zadanie aproksymacji kodystrybucji stowarzyszonych z odwrotnościami jakobianu pseudoodwrotnego i rozszerzonego. W ostatniej części rozdziału zostały opisane dwie metody rozwiązywania zadań wariacyjnych. Jest to klasyczna metoda Eulera–Lagrange’a oraz metoda Ritza, która jest metodą bezpośrednią. Pozwalają one na obliczyć funkcję rozszerzającą minimalizującą odpowiedni funkcjonal błędu aproksymacji. Inny rodzaj algorytmów aproksymujących omawia rozdział trzeci. Opisuje on zastosowanie metody homotopii do rozwiązania zadania reprodukcji trajektorii stanu lub wyjścia robota. Otrzymane za pomocą wyprowadzonych algorytmów funkcje sterujące są aproksymacją funkcji sterujących realizujących zadaną trajektorię. Rozdział czwarty zawiera przykłady obliczeń numerycznych dla zadania aproksymacji algorytmu jakobianu pseudoodwrotnego przez algorytm jakobianu rozszerzonego. Symulacje zostały wykonane dla redundantnych manipulatorów stacjonarnych: prostego manipulatora 3R bez konfiguracji osobliwych, manipulatora typu TTR i manipulatora przemysłowego – POLYCRANK o 7 stopniach swobody. W omawianym rozdziale pojawił się również przykład syntezy algorytmów aproksymacyjnych dla robota mobilnego – monocykla. Rozdział zawiera porównanie metody wariacyjnej i metody geometrycznej. Na koniec omówione są aspekty obliczeniowe. Przykłady rozwiązywania zadania reprodukcji trajektorii stanu i zadania reprodukcji trajektorii wyjścia przedstawione są w rozdziale piątym. Efektywność działania algorytmów została pokazana dla modeli: podwójnego integratora, kinematyki monocykla, dynamiki łyżwiarza Czapłygina oraz kinematyki kuli kinematycznej. Pracę podsumowuje rozdział szósty, w którym także nakreślono kierunki dalszych badań.

Badania prowadzone w tej pracy były wsparte ze środków Fundacji na rzecz Nauki Polskiej, Politechniki Wrocławskiej w ramach grantów statutowych oraz ze środków MNiSzW na naukę w latach 2010–2012 jako projekt promotorski.

Rozdział 2

Aproksymacja jakobianowych algorytmów kinematyki odwrotnej

W tym rozdziale będziemy rozważać problem syntezy jakobianowych algorytmów kinematyki odwrotnej. Celem będzie zdefiniowanie algorytmu typu jakobianu rozszerzonego aproksymującego algorytm typu jakobianu pseudoodwrotnego. Przedstawimy dwa podejścia: pierwszy korzystający z rachunku wariacyjnego, drugi z geometrii różniczkowej. Dla metod wariacyjnych wyprowadzone zostaną dwa kryteria aproksymacji zależne od funkcji rozszerzającej. Rozważymy również różne sposoby reprezentacji kinematyki, we współrzędnych zadaniowych i bez współrzędnych, za pomocą odwzorowania w $SE(3)$.

2.1 Preliminaria

Kinematyka manipulatora o n stopniach swobody zdefiniowana jest jako przekształcenie

$$\mathcal{K}: \mathbb{R}^n \rightarrow SE(3), \quad Y = \mathcal{K}(x) = \begin{bmatrix} R(x) & T(x) \\ 0 & 1 \end{bmatrix}, \quad (2.1)$$

przyporządkowujące każdej konfiguracji przegubowej $x \in \mathbb{R}^n$ odpowiednią pozycję i orientację końca efektora. Pozycja i orientacja należą do specjalnej grupy euklidesowej pozwalającej opisać ruch ciała sztywnego $SE(3) \cong SO(3) \times \mathbb{R}^3$, gdzie $SO(3)$ jest grupą obrotów. Przy zadanej kinematyce manipulatora (2.1) i pozycji efektora w przestrzeni zadaniowej $Y_d \in SE(3)$, odwrotne zadanie kinematyki polega na znalezieniu konfiguracji x_d , dla której

$$\mathcal{K}(x_d) = Y_d.$$

Używając współrzędnych przegubowych x i wybierając pewien układ współrzędnych zadaniowych otrzymujemy reprezentację kinematyki we współrzędnych

$$k: \mathbb{R}^n \rightarrow \mathbb{R}^m, \quad y = k(x) = (k_1(x), \dots, k_m(x))^T, \quad (2.2)$$

transformującą przestrzeń konfiguracyjną manipulatora $x \in \mathbb{R}^n$ w przestrzeń zadaniową $y \in \mathbb{R}^m$. Niech $J(x) = \frac{\partial k(x)}{\partial x}$ oznacza jakobian analityczny manipulatora. Konfiguracja x jest regularna jeśli jakobian $J(x)$ jest suriekcją, w przeciwnym przypadku konfiguracja jest osobliwa. Mając daną kinematykę (2.2) i zadaną konfigurację końca efektora możemy zdefiniować odwrotne zadanie kinematyki: wyznaczyć konfigurację przegubową x_d , dla której $k(x_d) = y_d$. Do rozwiązania odwrotnego zadania kinematyki będziemy posługiwali się algorytmem jakobianowym, którego wyprowadzenie korzysta z metody homotopii.

Mając daną konfigurację początkową x_0 , wybieramy gładką krzywą $x(t)$ w przestrzeni konfiguracyjnej przechodzącą przez x_0 , w taki sposób, by błąd w przestrzeni zadaniowej $e(t) = k(x(t)) - y_d$ wzdłuż tej krzywej zanikał eksponencjalnie ze współczynnikiem $\gamma > 0$,

$$\dot{e}(t) = J(x(t))\dot{x}(t) = -\gamma e(t). \quad (2.3)$$

Różniczkując powyższe równanie otrzymujemy równanie Ważewskiego–Dawidenki

$$J(x(t))\frac{dx(t)}{dt} = -\gamma(k(x(t)) - y_d). \quad (2.4)$$

Założmy, że $J^\#(x) \in \mathbb{R}^{n \times m}$ jest prawostronną odwrotnością jacobianu tzn. $J(x)J^\#(x) = I_m$. Ostatecznie, jacobianowy algorytm kinematyki odwrotnej jest zdefiniowany przez układ dynamiczny

$$\dot{x}(t) = -\gamma J^\#(x(t))e(t), \quad (2.5)$$

którego trajektoria prowadzi do rozwiązania $x_d = \lim_{t \rightarrow +\infty} x(t)$. Zauważmy, że każdy algorytm oparty na prawostronnej odwrotności jacobianu będzie posiadał taką samą zbieżność błędu, zależną od γ . Jednakże, zbieżność w przestrzeni konfiguracyjnej manipulatora, wyrażona przez (2.5), będzie różna, w zależności od obranego algorytmu.

Jacobianowe algorytmy kinematyki odwrotnej najczęściej opierają się na dwóch rodzajach odwrotności jacobianu: pseudoodwrotności i odwrotności typu jacobianu rozszerzonego. Pseudoodwrotność jacobianu opisuje rozwiązanie odwrotnego zadania kinematyki układu

$$\dot{y} = J(x)\dot{x}$$

spełniające warunek minimalizacji formy kwadratowej [59]

$$\mathcal{X}(x) = \frac{1}{2}\dot{x}^\top \dot{x}. \quad (2.6)$$

Aby wyznaczyć pseudoodwrotność, należy rozwiązać zadanie optymalizacji funkcji (2.6) z ograniczeniami równościowymi. Po rozwiązaniu tego zadania otrzymujemy pseudoodwrotność jacobianu

$$J^{P\#}(x) = J^\top(x) (J(x)J^\top(x))^{-1}. \quad (2.7)$$

Alternatywnie, stosuje się odwrotność typu jacobianu rozszerzonego zdefiniowaną w następujący sposób. Mając daną kinematykę w postaci (2.2), wybieramy odwzorowanie rozszerzające

$$h: \mathbb{R}^n \rightarrow \mathbb{R}^s, \quad h(x) = (h_1(x), \dots, h_s(x)), \quad s = n - m. \quad (2.8)$$

Korzystając z (2.8), możemy zdefiniować kinematykę rozszerzoną

$$l = (k, h): \mathbb{R}^n \rightarrow \mathbb{R}^n, \quad \bar{y} = l(x),$$

będącą przekształceniem przestrzeni $\mathbb{R}^n$ w $\mathbb{R}^n$ oraz jacobian rozszerzony

$$\bar{J}(x) = \frac{\partial l(x)}{\partial x}.$$

Ostatecznie, odwrotność typu jacobianu rozszerzonego przyjmuje postać

$$J^{E\#}(x) = \bar{J}^{-1}(x) \Big|_{m \text{ pierwszych kolumn}} \quad (2.9)$$

i jest prawostronną odwrotnością jacobianu

$$J(x)J^{E\#}(x) = I_m,$$

jednocześnie spełniającą warunek anihilacji

$$\frac{\partial h(x)}{\partial x} J^{E\#}(x) = 0.$$

Każdy z wyżej wymienionych algorytmów posiada swoje zalety:

- Algorytm typu jacobianu pseudoodwrotnego charakteryzuje się szybką zbieżnością i lokalną minimalizacją prędkości w przegubach w trakcie ruchu.
- Algorytm typu jacobianu rozszerzonego jest powtarzalny, tzn. przekształca zamknięte krzywe w przestrzeni zadaniowej w zamknięte krzywe w przestrzeni konfiguracyjnej.

Idea aproksymacji jacobianu pseudoodwrotnego przez jacobian rozszerzony pozwala stworzyć nowy algorytm jacobianowy charakteryzujący się zaletami obu algorytmów składowych. Klasyczne sformułowanie zadania aproksymacji sprowadza się do problemu optymalizacji, który może zostać rozwiązane za pomocą metod rachunku wariacyjnego. Alternatywnie, kolumny macierzy definiującej odwrotności jacobianów (2.7) i (2.9) w równaniu (2.5) mogą być potraktowane jako pola wektorowe w $\mathbb{R}^n$. Z każdym z tych algorytmów można więc stowarzyszyć dystrybucję

$$\begin{aligned}\mathcal{D}_P &= \text{span}_{C^\infty(\mathbb{R}^n)} \left\{ J_1^{P\#}(x), J_2^{P\#}(x), \dots, J_m^{P\#}(x) \right\}, \\ \mathcal{D}_E &= \text{span}_{C^\infty(\mathbb{R}^n)} \left\{ J_1^{E\#}(x), J_2^{E\#}(x), \dots, J_m^{E\#}(x) \right\}.\end{aligned}$$

W każdej konfiguracji $x \in \mathbb{R}^n$ powyższe dystrybucje definiują m -wymiarową przestrzeń liniową dopuszczalnych kierunków ruchu. Dystrybucja jest całkowalna, jeśli przez każdy punkt x przechodzi m -wymiarowa rozmaitość, której przestrzeń styczna jest zdefiniowana przez tę dystrybucję. W ogólnym przypadku, dystrybucja $\mathcal{D}_P$ nie jest całkowalna, w przeciwieństwie do $\mathcal{D}_E$, która posiada tę cechę dzięki własności anihilacji. Twierdzenie Frobeniusa orzeka, że dystrybucja jest całkowalna, jeśli jest involutywna, co oznacza że nawias Liego dowolnych pól wektorowych z dystrybucji należy do tej dystrybucji. Całkowalność stowarzyszonych dystrybucji zapewnia powtarzalność algorytmu jacobianowego.

1-formy różniczkowe anihilujące powyższe dystrybucje definiują parę kodystrybucji

$$\begin{aligned}\mathcal{D}_P^\perp &= \text{span}_{C^\infty(\mathbb{R}^n)} \left\{ \Omega_1(x), \Omega_2(x), \dots, \Omega_s(x) \right\}, \\ \mathcal{D}_E^\perp &= \text{span}_{C^\infty(\mathbb{R}^n)} \left\{ dh_1(x), dh_2(x), \dots, dh_s(x) \right\},\end{aligned}$$

gdzie $\Omega_i(x)J_j^{P\#}(x) = 0$, dla każdego $i = 1, \dots, s$ i $j = 1, \dots, m$. Z definicji dystrybucji i (2.7) wynika, że 1-formy rozpinające $\mathcal{D}_P^\perp$ anihilują jacobian transponowany $J^\top(x)$. Opis algorytmów jacobianowych za pomocą dystrybucji lub kodystrybucji jest równoważny. Jednakże, dla niskiego stopnia redundancji s , efektywniejsze może być zastosowanie kodystrybucji.

2.2 Metoda wariacyjna

2.2.1 Funkcjonał błędu Robertsa i Maciejewskiego

W tym podrozdziale zostanie przedstawione zadanie aproksymacji zdefiniowane przez R. Robertsa i A. Maciejewskiego w [38, 39]. W tych pracach, autorzy przedstawili sposób umożliwiający zbudowanie powtarzalnego algorytmu kinematyki odwrotnej, przypominający w działaniu inny algorytm, np. algorytm jacobianu pseudoodwrotnego.

Rozważmy prawostronną odwrotność $G(x)$ jacobianu $J(x)$ dla układu (2.2)

$$\dot{x} = G(x)\dot{y}.$$

Roberts i Maciejewski w [39] zauważyli, że każdą prawostronną odwrotność jacobianu możemy przedstawić jako

$$G(x) = [g_1(x), \dots, g_m(x)] = J^{P\#}(x) + K(x)W^T(x), \quad (2.10)$$

gdzie $J^{P\#}(x)$ jest pseudoodwrotnością jacobianu bez osobliwości, $K(x)$ jest macierzą $n \times s$ o ortonormalnych kolumnach, $K^T(x)K(x) = I_s$, rozpinającą jądro jacobianu, $J(x)K(x) = 0$, a $W(x)$ jest macierzą, której kolumnami są odpowiednie wektory, które w sposób jednoznaczny definiują $G(x)$. Dobierając $W(x)$ w taki sposób, aby dystrybucja rozpięta przez kolumny macierzy $G(x)$ była inwolutywna otrzymamy powtarzalną odwrotność [43]. Korzystając z (2.10) łatwo pokazać, że

$$N_G(x) = J^T(x)W(x) - K(x)$$

jest macierzą o własności $N_G^T(x)G(x) = 0$, co oznacza że wiersze $N_G^T(x)$ są 1-formami anihilującymi $G(x)$. Wynika stąd, że dystrybucja rozpięta przez kolumny $G(x)$ jest inwolutywna wtedy i tylko wtedy, gdy nawiasy Liego $[g_i, g_j] = \left(\frac{\partial g_j}{\partial x}\right)g_i - \left(\frac{\partial g_i}{\partial x}\right)g_j$ kolumn macierzy $G(x)$ spełniają warunek

$$N_G^T(x)[g_i, g_j](x) = 0, \quad 1 \leq i \leq j \leq m. \quad (2.11)$$

Z faktu, że kolumny macierzy $G(x)$ zależą od elementów macierzy $W(x)$ wynika, że aby wyznaczyć macierz $W(x)$ należy rozwiązać układ równań różniczkowych cząstkowych (2.11). Ponieważ odwrotności posiadających cechę powtarzalności jest nieskończenie dużo, możliwe jest wybranie takiej, która będzie zapewniała spełnienie dodatkowych warunków. Jednym z możliwych warunków jest minimalizacja odległości między pewną zadaną odwrotnością $G_d(x)$ a powtarzalną odwrotnością $G_r(x)$ (na przykład, spełniającą równania (2.11)). Wówczas, miarą odległości będzie

$$\int_{\mathbb{Q}} \|G_r(x) - G_d(x)\|_F^2 dx, \quad (2.12)$$

gdzie $\|M\|_F = \sqrt{\text{tr}(MM^T)}$ jest normą Frobeniusa, a $\mathbb{Q}$ jest obszarem aproksymacji z założenia złożonym z konfiguracji regularnych. Równanie (2.12) jest miarą odległości między dwiema odwrotnościami. Korzystając z (2.10) mamy

$$\|G_r(x) - G_d(x)\|_F = \|K(x)(W_r(x) - W_d(x))^T\|_F.$$

Ponieważ kolumny macierzy $K(x)$ są wektorami o długości jednostkowej, kryterium optymalizacji sprowadza się do funkcjonału

$$\int_{\mathbb{Q}} \|G_r(x) - G_d(x)\|_F^2 dx = \int_{\mathbb{Q}} \|W_r(x) - W_d(x)\|_F^2 dx. \quad (2.13)$$

W praktyce, rozwiązanie cząstkowych równań różniczkowych (2.11) może być niemożliwe, co czyni problem nierozwiązalnym. Z tego powodu, w celu uzyskania rozwiązania od razu wybieramy $G_r(x)$ w postaci zapewniającej powtarzalność. Własność tą posiada odwrotność typu jakobianu rozszerzonego

$$G_r(x) = J^{E\#}(x) = \left[\frac{J(x)}{\frac{\partial h(x)}{\partial x}} \right]^{-1} \Bigg|_{m \text{ pierwszych kolumn}}, \quad (2.14)$$

gdzie $\frac{\partial h(x)}{\partial x}$ jest macierzą Jacobiego odwzorowania rozszerzającego $h(x)$. Załóżmy, że

$$G_d(x) = J^{P\#}(x) + K(x)W_d^\top(x), \quad (2.15)$$

$$G_r(x) = J^{P\#}(x) + K(x)W_r^\top(x). \quad (2.16)$$

Ponieważ chcemy, żeby $G_r(x)$ było odwrotnością typu jakobianu rozszerzonego to

$$J(x)G_r(x) = I_m, \quad (2.17)$$

$$\frac{\partial h(x)}{\partial x} G_r(x) = 0. \quad (2.18)$$

Warunek (2.17) jest spełniony automatycznie, ponieważ $J(x)J^{P\#}(x) = I_m$ i $J(x)K(x) = 0$. Z (2.18) mamy

$$\frac{\partial h(x)}{\partial x} G_r(x) = \frac{\partial h(x)}{\partial x} J^{P\#}(x) + \frac{\partial h(x)}{\partial x} K(x)W_r^\top(x) = 0.$$

Zatem

$$W_r^\top(x) = - \left(\frac{\partial h(x)}{\partial x} K(x) \right)^{-1} \frac{\partial h(x)}{\partial x} J^{P\#}(x),$$

co podstawiając do (2.16) otrzymujemy

$$G_r(x) = J^{P\#}(x) - K(x) \left(\frac{\partial h(x)}{\partial x} K(x) \right)^{-1} \frac{\partial h(x)}{\partial x} J^{P\#}(x).$$

Z drugiej strony, z (2.15) możemy wyliczyć

$$J^{P\#}(x) = G_d(x) - K(x)W_d^\top(x).$$

Wówczas

$$\begin{aligned} G_r(x) = G_d(x) - K(x)W_d^\top(x) - K(x) \left(\frac{\partial h(x)}{\partial x} K(x) \right)^{-1} \frac{\partial h(x)}{\partial x} G_d(x) + \\ \underbrace{K(x) \left(\frac{\partial h(x)}{\partial x} K(x) \right)^{-1} \frac{\partial h(x)}{\partial x} K(x)W_d^\top(x)}_{K(x)W_d^\top(x)}. \end{aligned}$$

Ostatecznie, zależność pozwalająca wyznaczyć nam powtarzalny jakobian wyraża się poprzez

$$G_r(x) = G_d(x) - K(x) \left(\frac{\partial h(x)}{\partial x} K(x) \right)^{-1} \frac{\partial h(x)}{\partial x} G_d(x),$$

zatem różnica

$$G_r(x) - G_d(x) = -K(x) \left(\frac{\partial h(x)}{\partial x} K(x) \right)^{-1} \frac{\partial h(x)}{\partial x} G_d(x). \quad (2.19)$$

Wstawiając (2.19) do (2.13) otrzymujemy kryterium optymalizacji

$$\int_{\mathbb{Q}} \|G_r(x) - G_d(x)\|_F^2 dx = \int_{\mathbb{Q}} \left\| \left(\frac{\partial h(x)}{\partial x} K(x) \right)^{-1} \frac{\partial h(x)}{\partial x} G_d(x) \right\|_F^2 dx. \quad (2.20)$$

Na potrzeby pracy przyjmijmy, że odwrotnością $G_d(x)$ nie posiadającą własności powtarzalności jest jacobian pseudoodwrotny $J^{P\#}(x)$, założmy także, że stopień redundancji manipulatora jest równy s . Wtedy, zadanie aproksymacji polega na znalezieniu odwzorowania rozszerzającego minimalizującego błąd aproksymacji

$$\mathcal{E}(h) = \int_{\mathbb{Q}} \|J^{P\#}(x) - J^{E\#}(x)\|_F^2 dx,$$

gdzie $\mathbb{Q} \subset \mathbb{R}^n$ jest zawarte w regionie konfiguracji regularnych. Dla tak postawionego zadania, z wzoru (2.20) wynika, że błąd aproksymacji jest równy

$$\mathcal{E}(h) = \int_{\mathbb{Q}} \text{tr} \left(\left(\frac{\partial h(x)}{\partial x} K(x) \right)^{-1} \frac{\partial h(x)}{\partial x} J^{P\#} J^{P\#T}(x) \left(\frac{\partial h(x)}{\partial x} \right)^T \left(\frac{\partial h(x)}{\partial x} K(x) \right)^{-T} \right) dx. \quad (2.21)$$

Aby otrzymać optymalną funkcję rozszerzającą, minimalizującą funkcjonal (2.21), należy rozwiązać układ nieliniowych cząstkowych równań różniczkowych, wynikających z równania Eulera–Lagrange’a. Rozwiązanie takiego układu równań jest bardzo trudne, a w przypadku realistycznych manipulatorów przemysłowych wręcz niemożliwe.

2.2.2 Interpretacja

Przyjrzyjmy się dokładniej matematycznej interpretacji aproksymacji algorytmów. W tym celu zdefiniujemy dwa algorytmy kinematyki odwrotnej

$$\begin{aligned} \dot{x}(t) &= \bar{X}(x(t)) = X(x(t))\alpha(x(t)), \\ \dot{y}(t) &= \bar{Y}(y(t)) = Y(y(t))\alpha(y(t)), \end{aligned} \quad (2.22)$$

gdzie $X(x) = [X_1(x), \dots, X_m(x)]$ i $Y(y) = [Y_1(y), \dots, Y_m(y)]$ są odwrotnościami jacobianu, błąd w przestrzeni zadaniowej $\alpha(x) = k(x) - y_d$, a y_d jest zadany położeniem i orientacją efektora. Obliczmy różnicę trajektorii w przestrzeni konfiguracyjnej

$$x(t) - y(t) = x(0) - y(0) + \int_0^t (\bar{X}(x(u)) - \bar{Y}(y(u))) du. \quad (2.23)$$

Aby wyznaczyć powyższą różnicę rozpiszmy

$$\begin{aligned} \bar{X}(x) - \bar{Y}(y) &= X(x)\alpha(x) - Y(y)\alpha(y) = \\ &= (X(x) - Y(x))\alpha(x) + (Y(x) - Y(y))\alpha(y) + Y(x)(\alpha(x) - \alpha(y)). \end{aligned}$$

Korzystając z lematu Hadamarda [18] mamy

$$f(x) - f(y) = \int_0^1 df(tx + (1-t)y) = \int_0^1 Df(tx + (1-t)y) dt(x-y).$$

Stąd

$$(Y(x) - Y(y))\alpha(y) = \int_0^1 DY(tx + (1-t)y)\alpha(y) dt(x-y), \quad (2.24)$$

$$Y(x)(\alpha(x) - \alpha(y)) = Y(x) \int_0^1 D\alpha(tx + (1-t)y) dt(x-y). \quad (2.25)$$

Założmy, że $x(0) = y(0)$. Podstawiając (2.24) i (2.25) do (2.23) otrzymujemy

$$\begin{aligned} x(t) - y(t) &= \int_0^t (X(x(u)) - Y(x(u)))\alpha(x(u)) du + \\ &\quad \int_0^t \int_0^1 DY(sx(u) + (1-s)y(u))\alpha(y(u)) ds(x(u) - y(u)) du + \\ &\quad \int_0^t Y(x(u)) \int_0^1 D\alpha(sx(u) + (1-s)y(u)) ds(x(u) - y(u)) du. \end{aligned}$$

Dalej

$$\begin{aligned} \|x(t) - y(t)\| &\leq \int_0^t \|(X(x(u)) - Y(x(u)))\alpha(x(u))\| du + \\ &\quad \int_0^t \int_0^1 \|DY(sx(u) + (1-s)y(u))\alpha(y(u))\| ds\|x(u) - y(u)\| du + \\ &\quad \int_0^t \|Y(x(u))\| \int_0^1 \|D\alpha(sx(u) + (1-s)y(u))\| ds\|x(u) - y(u)\| du. \end{aligned}$$

Rozważmy

$$\begin{aligned} \|(X(x(u)) - Y(x(u)))\alpha(x(u))\|^2 &= \alpha^\top(x)(X(x) - Y(x))^\top(X(x) - Y(x))\alpha(x) = \\ &= \text{tr}((X(x) - Y(x))\alpha(x))^\top(X(x) - Y(x))\alpha(x) = \|(X(x) - Y(x))\alpha(x)\|_F^2 \leq \\ &\leq \|X(x) - Y(x)\|_F^2 \|\alpha(x)\|_F^2 = \|X(x) - Y(x)\|_F^2 \|\alpha(x)\|^2, \end{aligned}$$

gdzie $\|\cdot\|$ jest normą euklidesową. Mając różnicę wzdłuż krzywej leżącej w obszarze $\mathbb{Q}$ będącym zwartym podzbiorem $\mathbb{R}^n$, możemy napisać

$$\begin{aligned} \int_0^t \|(X(x(u)) - Y(x(u)))\alpha(x(u))\| du &\leq \\ &\leq \int_{\mathbb{Q}} \|(X(x) - Y(x))\alpha(x)\| dx \leq \int_{\mathbb{Q}} \|X(x) - Y(x)\|_F \|\alpha(x)\| dx. \end{aligned}$$

Założmy, że nierówności

$$\|\alpha(x)\| \leq a, \quad \|DY(x)\alpha(x)\| \leq d, \quad \|D\alpha(x)\| \leq c, \quad \|Y(x)\| \leq b$$

są spełnione w $\mathbb{Q}$. Wówczas

$$\begin{aligned} \|x(t) - y(t)\| &\leq \\ &a \int_{\mathbb{Q}} \|X(x) - Y(x)\|_F dx + d \int_0^t \|x(u) - y(u)\| du + bc \int_0^t \|x(u) - y(u)\| du = \\ &a \int_{\mathbb{Q}} \|X(x) - Y(x)\|_F dx + \gamma \int_0^t \|x(u) - y(u)\| du, \\ &\gamma = d + bc. \end{aligned}$$

Na podstawie nierówności Gronwalla [18] możemy napisać

$$\|x(t) - y(t)\| \leq a \int_{\mathbb{Q}} \|X(x) - Y(x)\|_F dx \exp(\gamma t).$$

Zatem, jeśli dla skończonych t miara odległości dwóch odwrotności jacobianu $\int_{\mathbb{Q}} \|X(x) - Y(x)\|_F dx$, jest mała to także różnica trajektorii w przestrzeni konfiguracyjnej jest mała. Wobec powyższego możemy sformułować następujące twierdzenie.

Twierdzenie 2.2.1 *Niech $X(x)$ i $Y(x)$ oznaczają odwrotności jacobianów, a $x(t)$, $y(t)$ trajektorie w przestrzeni konfiguracyjnej wyznaczone przez odpowiednie jacobianowe algorytmy kinematyki odwrotnej (2.22). Wówczas, dla pewnych liczb $a, \gamma \geq 0$ prawdziwa jest nierówność*

$$\|x(t) - y(t)\| \leq a \int_{\mathbb{Q}} \|X(x) - Y(x)\|_F dx \exp(\gamma t).$$

2.2.3 Funkcjonał błędu I

Kinematyka we współrzędnych

Aby rozwiązać zadanie aproksymacji przy pomocy funkcjonału błędu (2.21) zdefiniowanego przez Roberta i Maciejewskiego, należy rozwiązać układ nieliniowych cząstkowych równań różniczkowych. Rozwiązanie tak określonego układu równań jest zadaniem bardzo trudnym, nawet dla manipulatora o kilku stopniach swobody. W przypadku bardziej skomplikowanych manipulatorów rozwiązanie zadania optymalizacji staje się wręcz niemożliwe, biorąc pod uwagę nie tylko metody analityczne, ale także dostępne metody numeryczne. Dlatego, w niniejszym rozdziale pracy proponujemy modyfikację metody Roberta i Maciejewskiego, która prowadzi do sformułowania nowego błędu aproksymacji.

Funkcjonał błędu I opiera się na naturalnym zanurzeniu macierzy odwrotności jacobianu $J^{E\#}(x)$ i $J^{P\#}(x)$ w parę macierzy wymiaru $n \times n$, które pozwala zdefiniować nowy błąd aproksymacji [53, 56]. Zauważmy, że poza konfiguracjami osobliwymi $J^{E\#}(x)$ jest zawarty w odwrotności jacobianu rozszerzonego

$$E_1(x) = \begin{bmatrix} J(x) \\ \frac{\partial h(x)}{\partial x} \end{bmatrix}^{-1} = \begin{bmatrix} J^{E\#}(x) & Q(x) \end{bmatrix}, \quad (2.26)$$

gdzie $Q(x)$ jest pewną macierzą dopełniającą wymiaru $n \times (n - m)$. Podobnie, dla $J^{P\#}(x)$ można zaproponować zanurzenie

$$E_2(x) = \begin{bmatrix} J(x) \\ K^T(x) \end{bmatrix}^{-1} = \begin{bmatrix} J^{P\#}(x) & K(x) \end{bmatrix}, \quad (2.27)$$

gdzie kolumny $K(x)$ rozpinają jądro jacobianu ($J(x)K(x) = 0$), jednocześnie są ortogonalne $K^\top(x)K(x) = I_s$, $s = n - m$. Niech $\mathbb{Q} \subset \mathbb{R}^n$ oznacza podzbiór konfiguracji regularnych manipulatora, wówczas za pomocą (2.26) i (2.27) możemy zdefiniować błąd aproksymacji jako miarę odległości między $J^{P\#}(x)$ i $J^{E\#}(x)$ odniesioną do macierzy $E_1(x)$ i $E_2(x)$

$$\mathcal{E}_1(h) = \int_{\mathbb{Q}} \|E_1^{-1}(x)E_2(x) - I_n\|_F^2 m(x) dx, \quad (2.28)$$

gdzie $m(x) = \sqrt{\det(J(x)J^\top(x))}$ jest funkcją manipulowalności konfiguracji x , a $m(x)dx$ pełni rolę formy objętości. Podstawiając (2.26) i (2.27) do (2.28) otrzymujemy

$$\begin{aligned} \mathcal{E}_1(h) = \int_{\mathbb{Q}} & \left\| \begin{bmatrix} J(x)J^{P\#}(x) & J(x)K(x) \\ \frac{\partial h(x)}{\partial x} J^{P\#}(x) & \frac{\partial h(x)}{\partial x} K(x) \end{bmatrix} - I_n \right\|_F^2 m(x) dx = \\ & \int_{\mathbb{Q}} \text{tr} \left(\begin{bmatrix} J(x)J^{P\#}(x) - I_m & J(x)K(x) \\ \frac{\partial h(x)}{\partial x} J^{P\#}(x) & \frac{\partial h(x)}{\partial x} K(x) - I_s \end{bmatrix} \right. \\ & \quad \left. \times \begin{bmatrix} J(x)J^{P\#}(x) - I_m & J(x)K(x) \\ \frac{\partial h(x)}{\partial x} J^{P\#}(x) & \frac{\partial h(x)}{\partial x} K(x) - I_s \end{bmatrix}^\top \right) m(x) dx. \end{aligned}$$

Po przekształceniach, błąd aproksymacji przyjmuje postać

$$\mathcal{E}_1(h) = \int_{\mathbb{Q}} \text{tr} \left(\frac{\partial h(x)}{\partial x} P(x) \left(\frac{\partial h(x)}{\partial x} \right)^\top - 2 \frac{\partial h(x)}{\partial x} K(x) - I_s \right) m(x) dx, \quad (2.29)$$

gdzie

$$P(x) = J^{P\#}(x)J^{P\#T}(x) + K(x)K^\top(x) = (J^\top(x)J(x) + K(x)K^\top(x))^{-1}.$$

Kinematyka wyrażona w $SE(3)$

Niech kinematyka manipulatora będzie przekształceniem postaci (2.1). Wówczas, odwrotne zadanie kinematyki sprowadza się do znalezienia konfiguracji przegubowej x_d , dla której koniec efektora przyjmuje zadane położenie i orientację $Y_d \in SE(3)$, tak że $\mathcal{K}(x_d) = Y_d$.

Załóżmy, że $Y_d = \begin{bmatrix} R_d & T_d \\ 0 & 1 \end{bmatrix} \in SE(3)$, a $x(t)$ jest trajektorią w przestrzeni konfiguracyjnej. Wówczas, możemy zdefiniować błąd w postaci

$$E(t) = (\log(R(x(t))R_d^\top), T(x(t)) - T_d) = (E_1(t), e_2(t)), \quad (2.30)$$

gdzie logarytm macierzy obrotu $\log R = \frac{\phi}{2\sin\phi} (R - R^\top)$, zaś $0 \leq \phi < \pi$ spełnia równanie $1 + 2\cos\phi = \text{tr } R$. Błąd (2.30) należy do algebry Liego $se(3) \cong so(3) \times \mathbb{R}^3$, grupy $SE(3)$.

Stosując standardowy izomorfizm $v \mapsto [v] = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}$ między $\mathbb{R}^3$ i algebrą

Liego $so(3)$, możemy przedstawić błąd orientacji za pomocą wektora $e_1 \in \mathbb{R}^3$, w taki sposób, że $E_1(t) = [e_1(t)]$. Wówczas, błąd (2.30) będzie równoważny wektorowi błędu $e = (e_1^\top, e_2^\top)^\top \in \mathbb{R}^6$. Równoważnie do (2.3) możemy zdefiniować reprezentację błędu dla kinematyki wyrażonej w $SE(3)$.

Lemat 2.2.2 *Ewolucja błędu w przestrzeni zadaniowej jest opisana równaniem różniczkowym*

$$\dot{e}(t) = \begin{bmatrix} P_d(x(t)) & 0 \\ 0 & I_3 \end{bmatrix} \begin{bmatrix} J_R(x(t)) \\ J_T(x(t)) \end{bmatrix} \dot{x}(t) = J_{Md}(x(t))\dot{x}(t) = -\gamma e(t), \quad (2.31)$$

gdzie

$$P_d(x(t)) = I_3 - \frac{1}{2} [e_1(t)] + \left(\frac{1}{\phi^2(t)} + \frac{\sin \phi(t)}{2\phi(t)(\cos \phi(t) - 1)} \right) [e_1(t)]^2, \quad (2.32)$$

$$[e_1(t)] = \log(R(x(t))R_d^\top), \quad 1 + 2 \cos \phi(t) = \text{tr}(R(x(t))R_d^\top),$$

oraz

$$J_R(x) = [J_{R1}(x), \dots, J_{Rn}(x)], \quad [J_{Ri}(x)] = \frac{\partial R(x)}{\partial x_i} R^\top(x), \quad J_T(x) = \frac{\partial T(x)}{\partial x}. \quad (2.33)$$

□

Indeks dolny d w równaniach (2.31) i (2.32) oznacza zależność od zadanej macierzy rotacji R_d , indeksy R i T odnoszą się odpowiednio do rotacji i translacji.

Dowód Z definicji $R(x(t))R_d^\top = \exp E_1(t)$, tak więc

$$\left(\frac{d}{dt} \exp(E_1(t)) \right) \exp(-E_1(t)) = \dot{R}(x(t))R_d^\top (R(x(t))R_d^\top)^\top = \dot{R}(x(t))R^\top(x(t)). \quad (2.34)$$

Korzystając z formuły Hausdorffa [42] możemy stwierdzić, że lewa strona powyższego równania jest równa $[P_d^{-1}(x(t))\dot{e}_1(t)]$ dla $P_d(x(t))$ określonego przez (2.32). Z drugiej strony, rozpisując prawą stronę równania (2.34) otrzymujemy

$$\dot{R}(x(t))R^\top(x(t)) = \sum_{i=1}^n \frac{\partial R(x)}{\partial x_i} R^\top(x) \dot{x}_i(t) = \left[\sum_{i=1}^n J_{Ri}(x(t)) \dot{x}_i(t) \right] = [J_R(x(t))\dot{x}(t)],$$

jak zdefiniowano w (2.33). Ostatecznie, otrzymaliśmy górny blok równania (2.31). Dzięki postaci błędu e_2 (2.30), wyprowadzenie dolnego bloku jest trywialne.

□

Należy zauważyć, że $J_{Md}(x)$ jest rodzajem jacobianu geometrycznego manipulatora. Mając dany jacobian (2.31), możemy obliczyć jacobian pseudoodwrotny $J_{Md}^{P\#}(x) = J_{Md}^\top(x) (J_{Md}(x)J_{Md}^\top(x))^{-1}$ oraz zdefiniować odpowiedni algorytm kinematyki odwrotnej. Pomimo tego, że jacobian $J_{Md}(x)$ zależy od pożądanej rotacji R_d , algorytmy jacobianu pseudoodwrotnego i jacobianu rozszerzonego są od niej niezależne. Niech $J_{RT}(x) = \begin{bmatrix} J_R(x) \\ J_T(x) \end{bmatrix}$.

Twierdzenie 2.2.3 *Układ dynamiczny związany z jacobianem pseudoodwrotnym $J_{Md}^{P\#}(x)$ przyjmuje postać*

$$\dot{x} = -\gamma J_{RT}^{P\#}(x)e,$$

gdzie $J_{RT}^{P\#}(x)$ oznacza pseudoodwrotność $J_{RT}(x)$.

□

Dowód Z (2.31) mamy

$$J_{Md}(x(t)) = \begin{bmatrix} P_d(x(t)) & 0 \\ 0 & I_3 \end{bmatrix} J_{RT}(x(t)).$$

Następnie, za pomocą powyższej tożsamości możemy obliczyć

$$J_{Md}^{P\#}(x(t)) = J_{Md}^\top(x) (J_{Md}(x) J_{Md}^\top(x))^{-1} = J_{RT}^{P\#}(x(t)) \begin{bmatrix} P_d^{-1}(t) & 0 \\ 0 & I_3 \end{bmatrix}.$$

Ale jako że

$$P_d^{-1}(t) = I_3 + \frac{1 - \cos \phi(t)}{\phi^2(t)} [e_1(t)] + \frac{\phi(t) - \sin \phi(t)}{\phi^3(t)} [e_1(t)]^2,$$

zobacz [43], oraz $[v]w = v \times w$, otrzymujemy $P_d^{-1}(t)e_1(t) = e_1(t)$, zatem $J_{Md}^{P\#}(x)e = J_{RT}^{P\#}(x)e$.

□

Podobne wnioskowanie możemy zastosować w odniesieniu do algorytmu jacobianu rozszerzonego. Zaczynamy od obrania przekształcenia rozszerzającego $h: \mathbb{R}^n \rightarrow \mathbb{R}^{n-6}$. Wtedy, odwrotność jacobianu rozszerzonego (2.9)

$$J_{Md}^{E\#}(x) = \left[J_{Md}(x) \begin{bmatrix} \frac{\partial h(x)}{\partial x} \end{bmatrix}^{-1} \right] \Bigg|_{6 \text{ pierwszych kolumn}}$$

definiuje układ dynamiczny (2.5) odpowiadający algorytmowi typu jacobianu rozszerzonego. Przez analogię do Lematu 2.2.3 możemy powiedzieć, że

Twierdzenie 2.2.4 *Układ dynamiczny związany z odwrotnością jacobianu rozszerzonego $J_{Md}^{E\#}(x)$ przyjmuje postać*

$$\dot{x} = -\gamma J_{RT}^{E\#}(x)e, \quad (2.35)$$

gdzie $J_{RT}^{E\#}(x)$ jest jacobianem rozszerzonym stowarzyszonym z $J_{RT}(x)$.

□

Dowód Bez straty na ogólności, możemy założyć, że jacobian rozszerzony przyjmuje postać [58]

$$J_{Md}^{E\#}(x) = \begin{bmatrix} I_6 \\ -\left(\frac{\partial h(x)}{\partial x^2}\right)^{-1} \frac{\partial h(x)}{\partial x^1} \end{bmatrix} W_d^{-1}(x),$$

gdzie $x = (x^{1\top}, x^{2\top})^\top \in \mathbb{R}^6 \times \mathbb{R}^{n-6}$ i

$$W_d(x) = J_{Md}(x) \begin{bmatrix} I_6 \\ -\left(\frac{\partial h(x)}{\partial x^2}\right)^{-1} \frac{\partial h(x)}{\partial x^1} \end{bmatrix}.$$

Niech

$$W(x) = J_{RT}(x) \begin{bmatrix} I_6 \\ -\left(\frac{\partial h(x)}{\partial x^2}\right)^{-1} \frac{\partial h(x)}{\partial x^1} \end{bmatrix}.$$

A zatem

$$W_d^{-1}(x) = W^{-1}(x) \begin{bmatrix} P_d^{-1}(t) & 0 \\ 0 & I_3 \end{bmatrix}.$$

Lecz

$$\left[- \left(\frac{\partial h(x)}{\partial x^2} \right)^{-1} \frac{\partial h(x)}{\partial x^1} \right] W^{-1}(x) = J_{RT}^{E\#}(x),$$

zatem przywołując tożsamość $P_d^{-1}e_1 = e_1$, otrzymujemy układ dynamiczny opisany równaniem (2.35). $\square$

Reasumując Lematy 2.2.3 i 2.2.4, błąd aproksymacji dla przypadku, kiedy kinematyka jest opisana w $SE(3)$ możemy zdefiniować analogicznie do (2.29).

Niech kinematyka manipulatora będzie opisana równaniem (2.1) i niech przekształcenie rozszerzające $h(x)$ określa zadanie aproksymacji algorytmu jacobianu pseudoodwrotnego przez algorytm jacobianu rozszerzonego. Funkcjonał błędu dla tego problemu przybiera następująca postać

$$\mathcal{E}_2(h) = \int_{\mathbb{Q}} \text{tr} \left(\frac{\partial h(x)}{\partial x} P_{RT}(x) \left(\frac{\partial h(x)}{\partial x} \right)^T - 2 \frac{\partial h(x)}{\partial x} K_{RT}(x) - I_s \right) m_{RT}(x) dx, \quad (2.36)$$

gdzie

$$P_{RT}(x) = J_{RT}^{P\#}(x) J_{RT}^{P\#T}(x) + K_{RT}(x) K_{RT}^T(x),$$

kolumny macierzy $K_{RT}(x)$ stanowią ortogonalną bazę jądra jacobianu $J_{RT}(x)$, natomiast $m_{RT}(x) = \sqrt{\det(J_{RT}(x) J_{RT}^T(x))}$. Całkowanie w (2.36) odbywa się w podziorze regularnych konfiguracji manipulatora, w którym $J_{RT}^{P\#}(x)$ jest poprawnie zdefiniowane.

2.2.4 Funkcjonał błędu II

Alternatywne sformułowanie zadania aproksymacji polega na zdefiniowaniu błędu aproksymacji przy pomocy kodystrybucji określających odwrotności jacobianów. Funkcjonał błędu II posłuży nam do celów porównawczych jakości aproksymacji algorytmów otrzymanych metodą wariacyjną i metodą geometryczną.

Zauważmy, że każdy element kodystrybucji $\mathcal{D}_E^\perp = \text{span}_{C^\infty(\mathbb{R}^n)} \{dh_1(x), \dots, dh_s(x)\}$ określa tą samą odwrotność jacobianu rozszerzonego, dlatego możemy definiować odwrotność przez pochodną $Dh(x)$ odwzorowań rozszerzających kinematykę. Wówczas, błąd aproksymacji może być zdefiniowany jako

$$\mathcal{E}_3(h) = \int_{\mathbb{Q}} \|Dh(x) - \Omega(x)\|_F^2 dx, \quad (2.37)$$

gdzie

$$Dh(x) = \begin{bmatrix} \frac{\partial h_1(x)}{\partial x_1} & \dots & \frac{\partial h_1(x)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_s(x)}{\partial x_1} & \dots & \frac{\partial h_s(x)}{\partial x_n} \end{bmatrix} = \begin{bmatrix} dh_1(x) \\ dh_2(x) \\ \vdots \\ dh_s(x) \end{bmatrix},$$

natomiast wiersze macierzy Ω są utworzone przez 1-formy Ω_i anihilujące jacobian transponowany, $\Omega_i J^T(x) = 0$. $\mathbb{Q}$ oznacza podzbiór $\mathbb{R}^n$, na którym aproksymujemy. Funkcjonał błędu (2.37) powinien być minimalizowany ze względu na funkcje rozszerzające, w ten sposób zadanie aproksymacji sprowadza się do pewnego problemu rachunku wariacyjnego.

2.3 Metoda geometryczna

W metodzie geometrycznej, zadanie aproksymacji polega na skonstruowaniu całkowalnej kodystrybucji, która pokrywa się z zadaną kodystrybucją na pewnym obszarze przestrzeni konfiguracyjnej i wzdłuż określonych kierunków w $\mathbb{R}^n$ [54, 56]. Innymi słowy, szukamy kodystrybucji $\mathcal{D}_E^\perp$ skojarzonej z odwrotnością typu jacobianu rozszerzonego, która aproksymuje kodystrybucję $\mathcal{D}_P^\perp$ związaną z jacobianem pseudoodwrotnym. W tym celu, zdefiniujemy w przestrzeni $\mathbb{R}^n$ foliację z s -wymiarowymi liśćmi E_α , parametryzowaną przez $\alpha \in \mathbb{R}^m$ i z zerowym liściem odniesienia E_0 . Jak łatwo zauważyć, wymiar liści jest równy stopniowi redundancji kinematyki. Z foliacją związana jest homotopia $\Phi_t: \mathbb{R}^n \rightarrow \mathbb{R}^n$ dla $t \in [0, 1]$, taka że $\Phi_1 = \text{id}_{\mathbb{R}^n}$, $\Phi_0: \mathbb{R}^n \rightarrow E_0$, a dla $t, s \in \mathbb{R}$ mamy $\Phi_s \circ \Phi_t = \Phi_{st}$. Homotopia Φ_t zachowuje strukturę foliacji w tym sensie, że każdy liść E_α jest odwzorowany w liść $\Phi_t(E_\alpha) \subset E_{\alpha'}$. Homotopia definiuje pole wektorowe w $\mathbb{R}^n$

$$X(x) = \left. \frac{d\Phi_t(x)}{dt} \right|_{t=1}, \quad (2.38)$$

które określa charakterystyczny kierunek w $\mathbb{R}^n$. Co więcej, łatwo pokazać, że pole wektorowe (2.38) spełnia następującą tożsamość

$$t \frac{d\Phi_t(x)}{dt} = X(\Phi_t(x)), \quad (2.39)$$

w wyniku czego, homotopia $\Phi_t(x)$ może być traktowana jako strumień pola wektorowego $X(x)/t$, parametryzowanego czasem.

Następnie, wybierzmy specjalne współrzędne $x = (y, z) \in \mathbb{R}^n$, w taki sposób, że $z \in \mathbb{R}^s$ zmienia się wzdłuż liścia a $y \in \mathbb{R}^m$ jest stałe na każdym z liści. Dla tak wybranych współrzędnych, liść odniesienia jest zdefiniowany jako $E_0 = \{(0, z) | z \in \mathbb{R}^s\}$, a zatem $\Phi_0(y, z) = (0, z)$. Załóżmy, że na liściu E_α 1-formy rozpinające $\mathcal{D}_P^\perp$ są wyrażone za pomocą współrzędnych (y, z) , w taki sposób, że $\Omega_i|_{E_\alpha} = dz_i$, podczas gdy generatory kodystrybucji $\mathcal{D}_E^\perp$ przyjmują postać

$$\overline{\Omega}_i = \sum_{j=1}^s B_{ij}(x) dh_j(x), \quad (2.40)$$

gdzie $i = 1, \dots, s$ i $B(x) = [B_{ij}(x)] \neq 0$ jest macierzą funkcji gładkich. Zdefiniujemy teraz zadanie aproksymacji jacobianu pseudoodwrotnego przez odwrotność typu jacobianu rozszerzonego w następujący sposób. Znaleźć odwzorowanie rozszerzające $h(x) = (h_1(x), h_2(x), \dots, h_s(x))$, takie że $h(z, 0) = z$, a kodystrybucje $\mathcal{D}_P^\perp$ i $\mathcal{D}_E^\perp$ pokrywają się na liściach foliacji $\{E_\alpha\}$

$$\overline{\Omega}_i|_{E_\alpha} = \Omega_i|_{E_\alpha} = dz_i \quad (2.41)$$

i zgadzają się wzdłuż pola wektorowego $X(x)$

$$\Omega_i(x)X(x) = \overline{\Omega}_i(x)X(x), \quad (2.42)$$

dla każdego $i = 1, \dots, s$. Zauważmy, że

$$\overline{\Omega}_i(x) = \sum_{j=1}^s B_{ij}(x) dh_j(x) = \sum_{j=1}^s \sum_{k=1}^m B_{ij}(x) \frac{\partial h_j(x)}{\partial y_k} dy_k + \sum_{j=1}^s \sum_{l=1}^s B_{ij}(x) \frac{\partial h_j(x)}{\partial z_l} dz_l,$$

co w rezultacie daje

$$\overline{\Omega}_i|_{E_\alpha} = \sum_{j=1}^s \sum_{l=1}^s B_{ij}(x) \frac{\partial h_j(x)}{\partial z_l} dz_l.$$

W konsekwencji, formuła (2.41) jest równoznaczna z

$$B(x) \frac{\partial h(x)}{\partial z} = I_s. \quad (2.43)$$

Dalej, podstawiając $x = \Phi_t(y, z)$, $F_i(x) = \Omega_i(x)X(x)$ i korzystając z zależności (2.39) i (2.40), równanie (2.42) możemy zapisać w postaci

$$\begin{aligned} F_i(\Phi_t(y, z)) &= \sum_{j=1}^s B_{ij}(\Phi_t(y, z)) dh_j(\Phi_t(y, z)) X(\Phi_t(y, z)) = \\ &= t \sum_{j=1}^s B_{ij}(\Phi_t(y, z)) dh_j(\Phi_t(y, z)) \frac{d\Phi_t(y, z)}{dt} = \\ &= t \sum_{j=1}^s B_{ij}(\Phi_t(y, z)) \frac{dh_j(\Phi_t(y, z))}{dt} \end{aligned}$$

i wywnioskować, że

$$tB(\Phi_t(y, z)) \frac{dh \circ \Phi_t(y, z)}{dt} = F(\Phi_t(y, z)), \quad (2.44)$$

gdzie $F(x) = (F_1(x), \dots, F_s(x))$. Po podstawieniu $H(t, y, z) = h(\Phi_t(y, z))$ i niezbędnych przekształceniach z wykorzystaniem równań (2.43) i (2.44) otrzymujemy układ s równań różniczkowych, cząstkowych parametryzowanych współrzędną y

$$\frac{\partial H(t, y, z)}{\partial t} - \frac{\partial H(t, y, z)}{\partial z} \frac{F(\Phi_t(y, z))}{\partial t} = 0,$$

dla $i = 1, \dots, s$. Do rozwiązania tego układu możemy zastosować metodę charakterystyk [5]. Łatwo zauważyć, że $H(t, y, z(t)) = \text{const}$ wzdłuż rozwiązania $z(t)$ równania charakterystycznego

$$\frac{dz(t)}{dt} = -\frac{F(\Phi_t(y, z))}{t}, \quad z(0) = z_0. \quad (2.45)$$

Wiedząc o tym i korzystając z własności homotopii $\Phi_t(x)$ i z założenia, że na liściu odniesienia $h(0, z) = z$, dostajemy

$$\begin{aligned} H(1, y, z(1)) &= h(\Phi_1(y, z)) = h(y, z) = H(0, y, z(0)) \\ &= h(\Phi_0(y, z(0))) = h(0, z_0) = z_0, \end{aligned}$$

zatem $h(y, z) = z_0$.

Ostatnia zależność oznacza tyle, że aby wyznaczyć funkcję rozszerzającą $h(x)$, musimy rozwiązać następujące równanie

$$z(1) = z = \phi(1, y, h(y, z)) = \phi(1, y, z_0),$$

gdzie $\phi(t, y, z_0)$ jest strumieniem układu (2.45). Aby rozwiązać to zadanie, możemy posłużyć się całkowaniem wstecznym równania charakterystycznego (2.45) dla zadanego y , a następnie odzyskać oryginalną zmienną x .

Zauważmy, że warunki (2.41) i (2.42) implikują pokrywanie się kodystrybucji $\mathcal{D}_P^\perp$ i $\mathcal{D}_E^\perp$ na liściu E_α i wzdłuż trajektorii $\Phi_t(x)$ pola wektorowego (2.38) łączącego ten liść z liściem odniesienia E_0 . Otrzymane podzbiory przestrzeni konfiguracyjnej

$$P_\alpha = \bigcup \{\Phi_t(E_\alpha) \mid t \in [0, 1]\}$$

są nazywane stronicami.

2.4 Metody rozwiązań

2.4.1 Metoda pośrednia — rachunek wariacyjny

Zagadnieniem wariacyjnym nazywamy problem wyznaczenia wartości ekstremalnych funkcjonału [17]. W niniejszym podrozdziale przedstawimy warunek konieczny istnienia ekstremum funkcjonału zależnego od funkcji rozszerzającej $h(x)$ i zastosujemy go do rozwiązania zadania aproksymacji algorytmów.

Dla kompletności, przytoczmy sposób rozwiązania zadania wariacyjnego o pochodnych cząstkowych z [17]. Rozpatrzmy funkcjonal

$$J(h(\cdot)) = \int_{\mathbb{Q}} L(x_1, \dots, x_n, h(x), h_{x_1}, \dots, h_{x_n}) dx_1 \dots dx_n, \quad (2.46)$$

gdzie $L(\cdot)$ jest funkcją Lagrange'a, $h = h(x_1, \dots, x_n)$, natomiast h_{x_i} oznacza pochodną cząstkową pierwszego rzędu po x_i . Obliczmy wariację tego funkcjonału zakładając, że obszar całkowania $\mathbb{Q}$ nie ulega zmianie. Niech

$$h^*(x_1, \dots, x_n) = h(x_1, \dots, x_n) + \varepsilon \eta(x_1, \dots, x_n) + O(\varepsilon^2),$$

wówczas przez wariację funkcjonału (2.46) rozumiemy część liniową ze względu na ε z różnicy

$$J(h^*) - J(h).$$

W dalszej części wywodu dla uproszczenia zapisu uznajmy, że $h(x) = h(x_1, \dots, x_n)$, $\eta(x) = \eta(x_1, \dots, x_n)$, a $dx \approx dx_1 \dots dx_n$. Obliczając powyższą różnicę otrzymamy

$$\begin{aligned} J(h^*) - J(h) &= \int_{\mathbb{Q}} [L(x, h(x) + \varepsilon \eta(x), h_{x_1}(x) + \varepsilon \eta_{x_1}(x), \dots, h_{x_n}(x) + \varepsilon \eta_{x_n}(x)) \\ &\quad - L(x, h(x), h_{x_1}(x), \dots, h_{x_n}(x))] dx = \varepsilon \int_{\mathbb{Q}} \left[L_h \eta(x) + \sum_{k=1}^n L_{h_{x_k}} \eta_{x_k}(x) \right] dx + O(\varepsilon^2). \end{aligned}$$

Ostatecznie

$$\delta J = \varepsilon \int_{\mathbb{Q}} \left[L_h \eta(x) + \sum_{k=1}^n L_{h_{x_k}} \eta_{x_k}(x) \right] dx \quad (2.47)$$

jest wariacją funkcjonału (2.46). Przyjmując, że

$$L_{h_{x_k}} \eta_{x_k}(x) = \frac{\partial}{\partial x_k} (L_{h_{x_k}} \eta(x)) - \frac{\partial L_{h_{x_k}}}{\partial x_k} \eta(x)$$

i podstawiając do równania (2.47) dostaniemy

$$\delta J = \varepsilon \int_{\mathbb{Q}} \left[L_h - \sum_{k=1}^n \frac{\partial}{\partial x_k} L_{h_{x_k}} \right] \eta(x) dx + \varepsilon \int_{\mathbb{Q}} \sum_{k=1}^n \frac{\partial}{\partial x_k} (L_{h_{x_k}} \eta(x)) dx.$$

Całka

$$\varepsilon \int_{\mathbb{Q}} \sum_{k=1}^n \frac{\partial}{\partial x_k} (L_{h_{x_k}} \eta(x)) dx = \int_{\Gamma} L_{h_{x_1}} \eta(x) dx_1 \dots dx_n - \dots + (-1)^n L_{h_{x_n}} \eta(x) dx_1 \dots dx_{n-1}$$

na mocy wzoru Greena. Jeżeli przyjmiemy, że $\eta(x)$ zeruje się na brzegu Γ obszaru $\mathbb{Q}$, to całka ta równa się zeru. Tym samym, wzór dla wariacji sprowadza się do

$$\delta J = \varepsilon \int_{\mathbb{Q}} \left(L_h - \sum_{k=1}^n \frac{\partial}{\partial x_k} L_{h_{x_k}} \right) \eta(x) \, dx.$$

Z warunku koniecznego istnienia ekstremum funkcjonału $\delta J = 0$ dla wszystkich funkcji $\eta(x)$, wynika, że funkcja podcałkowa również musi zanikać

$$L_h - \sum_{k=1}^n \frac{\partial}{\partial x_k} L_{h_{x_k}} = 0. \quad (2.48)$$

Równanie (2.48) stanowi warunek konieczny na rozwiązanie zadania wariacyjnego i jest nazywane równaniem Eulera–Lagrange’a dla tego zagadnienia.

W przypadku, kiedy stopień redundancji $s > 1$, $h(x) = (h_1(x), \dots, h_s(x))$, należy rozwiązać układ równań Eulera–Lagrange’a

$$L_{h_i} - \sum_{k=1}^n \frac{\partial}{\partial x_k} \frac{\partial L}{\partial \frac{\partial h_i}{\partial x_k}} = 0,$$

dla $i = 1, \dots, s$.

Wyznaczenie minimalnego błędu aproksymacji sprowadza się do rozwiązania zadania wariacyjnego (2.29) dla przypadku kinematyki we współrzędnych, bądź (2.36) dla kinematyki wyrażonej w $SE(3)$. Korzystając z rachunku wariacyjnego oraz zakładając, że błąd jest całką funkcji Lagrange’a $L(x, \frac{\partial h(x)}{\partial x})$, możemy napisać równanie Eulera–Lagrange’a

$$\text{tr} \frac{\partial}{\partial x} \left(\frac{\partial L}{\partial \frac{\partial h_i(x)}{\partial x}} \right) = 0, \quad (2.49)$$

dla $i = 1, \dots, s$. Specyficzna postać tego równania wyprowadzonego dla (2.29) i (2.36) jest następująca

$$\begin{aligned} \text{tr} \frac{\partial}{\partial x} \left(m^{1,2}(x) P^{1,2}(x) \frac{\partial h_i(x)}{\partial x} \right) - \text{tr} \frac{\partial}{\partial x} (m^{1,2}(x) K_i^{1,2}(x)) = \\ \sum_{r=1}^n \left(\sum_{k=1}^n m^{1,2}(x) P_{kr}^{1,2}(x) \frac{\partial^2 h_i(x)}{\partial x_k \partial x_r} + \sum_{k=1}^n \frac{\partial (m^{1,2}(x) P_{kr}^{1,2}(x))}{\partial x_r} \frac{\partial h_i(x)}{\partial x_k} - \right. \\ \left. \frac{\partial (m^{1,2}(x) K_{ri}^{1,2}(x))}{\partial x_r} \right) = 0, \end{aligned}$$

dla $i = 1, \dots, s$, prawy, górny indeks odnosi się odpowiednio: 1 i równania (2.29) oraz 2 do (2.36). Dla każdego $i = 1, \dots, s$, warunki optymalności (2.49) sprowadzają się równania [53]

$$\Delta h_i = \text{div} K_i, \quad (2.50)$$

gdzie Δ jest operatorem Laplace’a, $\Delta = \sum_{j=1}^n \frac{\partial^2}{\partial x_j^2}$, a operator div oznacza dywergencję $\text{div} K_i(x) = \sum_{j=1}^n \frac{\partial K_{ij}(x)}{\partial x_j}$. Równanie różniczkowe (2.50) znane jest jako równanie Poissona. Z tego wynika, że aby wyznaczyć optymalne odwzorowanie rozszerzające $h(x)$, należy rozwiązać układ liniowych, eliptycznych równań różniczkowych cząstkowych.

W celu wyznaczenia minimalnego „Funkcjonału błędu II”, łatwo sprawdzić z (2.37), że

$$(dh_i(x) - \Omega_i(x)) (dh_i(x) - \Omega_i(x))^T = \sum_{j=1}^n \left(\frac{\partial h_i(x)}{\partial x_j} \right)^2 - 2 \sum_{j=1}^n \omega_j^i(x) \frac{\partial h_i(x)}{\partial x_j} + \sum_{j=1}^n (\omega_j^i(x))^2, \quad (2.51)$$

dla $i = 1, \dots, s$. Stąd, warunki optymalności przyjmują postać równań Eulera-Lagrange’a

$$\Delta h_i(x) = \text{div } \Omega_i(x), \quad (2.52)$$

gdzie Δ jest operatorem Laplace’a, a operator div oznacza dywergencję. Częstkowe równanie różniczkowe (2.52) jest równaniem Poissona. W przypadku, gdy prawa strona równania (2.52) jest równa zero, jego rozwiązaniem są funkcje harmoniczne. W rzeczywistych przypadkach układów robotycznych, nie jesteśmy w stanie rozwiązać równania Poissona, nawet korzystając z metod numerycznych. Dlatego, aby otrzymać rozwiązanie konieczne jest zastosowanie metod bezpośrednich rachunku wariacyjnego, np. metody Ritza, patrz podrozdział 2.4.2.

2.4.2 Metoda bezpośrednia — metoda Ritza

Podstawą metod bezpośrednich jest wykorzystanie tzw. minimalizujących ciągów funkcji. Jedną z najbardziej znanych metod jest metoda Ritza [17]. Dla uproszczenia wyводу, przyjmijmy stopień redundancji $s = 1$. Załóżmy, że poszukujemy minimum funkcjonału

$$J(h(\cdot)), \quad (2.53)$$

określonego na pewnej rozmaitości będącej obszarem całkowania w $\mathbb{R}^n$. Weźmy pod uwagę pewien ciąg funkcji bazowych $b(x) = (b_1(x), b_2(x), \dots)$, taki że

$$h(x) = a_1 b_1(x) + a_2 b_2(x) + \dots = a^T b(x)$$

spełnia warunki brzegowe tego funkcjonału. Zadanie polega na znalezieniu współczynników a_i dla których wartość funkcjonału (2.53) jest najmniejsza.

Tak więc, dzięki sparametryzowanej postaci $h(x)$, minimalizacja błędu aproksymacji

$$\mathcal{E}_{1,2}(h) = \int_{\mathbb{Q}} \text{tr} \left(\frac{\partial h(x)}{\partial x} P^{1,2}(x) \left(\frac{\partial h(x)}{\partial x} \right)^T - 2 \frac{\partial h(x)}{\partial x} K^{1,2}(x) - I_s \right) m^{1,2}(x) dx,$$

gdzie prawy górny indeks 1 odpowiada (2.29) i 2 analogicznie (2.36), ze względu na parametry a_i , sprowadza się do minimalizacji formy kwadratowej

$$\mathcal{E}_{1,2}(a) = a^T M^{1,2} a - 2a^T N^{1,2} + d^{1,2}, \quad (2.54)$$

gdzie

$$\begin{aligned} M^{1,2} &= \int_{\mathbb{Q}} \frac{\partial b(x)}{\partial x} P^{1,2}(x) \left(\frac{\partial b(x)}{\partial x} \right)^T m^{1,2}(x) dx, \\ N^{1,2} &= \int_{\mathbb{Q}} \frac{\partial b(x)}{\partial x} K^{1,2}(x) m^{1,2}(x) dx, \\ d^{1,2} &= \int_{\mathbb{Q}} m^{1,2}(x) I_s dx. \end{aligned} \quad (2.55)$$

Różniczkując równanie (2.54) i przyrównując do zera otrzymujemy

$$\frac{\partial F(a)}{\partial a} = 2M^{1,2}a - 2N^{1,2} = 0. \quad (2.56)$$

Pod warunkiem, że macierz $M^{1,2}$ jest nieosobliwa, optymalne odwzorowanie rozszerzające kinematykę jest równe

$$h(x) = a^{*\top} b(x), \quad (2.57)$$

gdzie

$$a^* = (M^{1,2})^{-1} N^{1,2}.$$

Aby, zastosować metodę Ritza do rozwiązania zadania aproksymacji (2.37), zakładamy, że odwzorowanie rozszerzające kinematykę, dla stopnia redundancji $s = 1$, ma postać $h(x) = a^\top b(x)$, gdzie $b(x) = [b_1(x), \dots, b_p(x)]$ jest wektorem funkcji bazowych i $a \in \mathbb{R}^p$ jest wektorem parametrów. Podstawiając $h(x)$ do równania (2.37) otrzymujemy błąd w postaci formy kwadratowej

$$\mathcal{E}_3(a) = a^\top M a - 2a^\top N + Q, \quad (2.58)$$

gdzie

$$\begin{aligned} M &= \int_{\mathbb{Q}} \frac{\partial b(x)}{\partial x} \left(\frac{\partial b(x)}{\partial x} \right)^\top dx, \\ N &= \int_{\mathbb{Q}} \frac{\partial b(x)}{\partial x} \Omega^\top(x) dx, \\ Q &= \int_{\mathbb{Q}} \Omega^\top(x) \Omega(x) dx. \end{aligned} \quad (2.59)$$

Różniczkując $\mathcal{E}_3(a)$ względem a i przyrównując do zera mamy

$$2Ma - 2N = 0. \quad (2.60)$$

Naturalnie, błąd (2.58) osiąga minimum globalne dla $a^* = M^{-1}N$.

Powyższe wzory można analogicznie rozszerzyć na przypadek wyższego stopnia redundancji, t.j. dla $s > 1$.

Rozdział 3

Zastosowanie metody homotopii do zadania reprodukcji trajektorii układów robotycznych

Ciekawym przykładem zastosowania algorytmów aproksymacyjnych są algorytmy oparte na metodzie homotopii. W rozdziale zostaną przedstawione algorytmy dedykowane zadaniom planowania trajektorii stanu [55] i wyjścia [21], oparte na metodzie homotopii. Wychodząc z metody homotopii, dla każdego z tych zadań wyprowadzone zostanie równanie różniczkowe cząstkowe, którego rozwiązaniem jest rodzina jednoparametrowych funkcji sterujących. W tej rodzinie, przy parametrze dążącym do nieskończoności, zawarte jest rozwiązanie zadania planowania trajektorii.

Założmy, że kinematyka nieholonomicznego robota mobilnego lub dynamika manipulatora bądź robota mobilnego jest przedstawiona jest za pomocą afinicznego układu sterowania z funkcją wyjścia

$$\begin{cases} \dot{x} = f(x) + G(x)u = f(x) + \sum_{i=1}^r g_i(x)u_i \\ y = w(x) = (w_1(x), w_2(x), \dots, w_m(x)), \end{cases} \quad (3.1)$$

gdzie $x \in \mathbb{R}^n$ są zmiennymi stanu, $u \in \mathbb{R}^r$ jest wektorem sterowań i $y \in \mathbb{R}^m$ są zmiennymi wyjściowymi. Zakładamy, że system (3.1) jest kwadratowy, w tym sensie, że liczba zmiennych wyjścia jest równa liczbie sterowań, $m = r$. Niech dopuszczalne funkcje sterujące u należą do pewnej przestrzeni $\mathcal{U}$ funkcji określonych na przedziale czasu $[0, T]$ oraz przyjmujących wartości w $\mathbb{R}^r$. Wszystkie pola wektorowe i funkcje układu (3.1) są gładkie. Mając dany stan początkowy x_0 i funkcję sterującą $u(t)$ możemy wyliczyć trajektorię stanu $x(t) = \varphi_{x_0, t}(u(\cdot))$ i trajektorię wyjścia $y(t) = w(x(t))$ układu sterowania (3.1).

Odwzorowanie wejście–wyjście układu (3.1), przy ustalonym x_0 , może być zdefiniowane jako

$$\mathcal{F}_{x_0}: \mathcal{U} \rightarrow \mathcal{Y}, \quad \mathcal{F}_{x_0}(u(\cdot))(t) = y(t). \quad (3.2)$$

Trajektorię wyjścia $y(\cdot) \in \mathcal{Y}$ będziemy nazywać reprodukowalną w x_0 wtedy, gdy istnieje funkcja sterująca $u(\cdot) \in \mathcal{U}$, taka że $y(\cdot) = \mathcal{F}_{x_0}(u(\cdot))$. Wyznamy odwzorowanie końcowe układu (3.1) w chwili t jako

$$\mathcal{F}_{x_0, t}: \mathcal{U} \rightarrow \mathbb{R}^m, \quad \mathcal{F}_{x_0, t}(u(\cdot)) = y(t) = w(x(t)).$$

Dla ograniczonych funkcji sterujących wyposażonych w topologię L_p , $p \geq 2$ odwzorowanie końcowe jest klasy C^1 [47]. Pochodną tego odwzorowania możemy wyznaczyć, posługując

się przybliżeniem liniowym układu (3.1) wzdłuż pary sterowanie–trajektoria $(u(t), x(t))$, w następujący sposób

$$D\mathcal{F}_{x_0,t}(u(\cdot))v(\cdot) = \frac{d}{d\alpha} \Big|_{\alpha=0} \mathcal{F}_{x_0,t}(u(\cdot) + \alpha v(\cdot)) = \frac{d}{d\alpha} \Big|_{\alpha=0} w(\varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot))) = \frac{\partial w(x(t))}{\partial x} \frac{d}{d\alpha} \Big|_{\alpha=0} \varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot)) \quad (3.3)$$

gdzie $\frac{\partial w(x(t))}{\partial x} = C(t)$. W celu wyznaczenia $\frac{d}{d\alpha} \Big|_{\alpha=0} \varphi_{x_0,t}(u(\cdot))$ zróżniczkujemy to wyrażenie względem t

$$\begin{aligned} \frac{d}{dt} \frac{d}{d\alpha} \Big|_{\alpha=0} \varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot)) &= \frac{d}{d\alpha} \Big|_{\alpha=0} \frac{d}{dt} \varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot)) = \\ \frac{d}{d\alpha} \Big|_{\alpha=0} (f(\varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot))) + G(\varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot)))(u(t) + \alpha v(t))) &= \\ \frac{\partial f(x(t))}{\partial x} \frac{d}{d\alpha} \Big|_{\alpha=0} \varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot)) + \frac{\partial G(x(t))u(t)}{\partial x} \frac{d}{d\alpha} \Big|_{\alpha=0} \varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot)) + G(x(t))v(t) &= \\ \frac{\partial(f(x(t)) + G(x(t))u(t))}{\partial x} \frac{d}{d\alpha} \Big|_{\alpha=0} \varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot)) + G(x(t))v(t). \end{aligned} \quad (3.4)$$

Niech $A(t) = \frac{\partial(f(x(t)) + G(x(t))u(t))}{\partial x}$, $B(t) = G(x(t))$. Zdefiniujmy zmienną pomocniczą $\xi(t) = \frac{d}{d\alpha} \Big|_{\alpha=0} \varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot))$. Wówczas, równanie (3.4) możemy zapisać jako następujące liniowe równanie różniczkowe, będące przybliżeniem liniowym układu (3.1) wzdłuż pary sterowanie–trajektoria $(u(t), x(t))$

$$\dot{\xi}(t) = A(t)\xi(t) + B(t)v(t), \quad (3.5)$$

z warunkiem początkowym $\xi(0) = 0$. Układ (3.5) nazywa się układem wariacyjnym stowarzyszonym z (3.1). Posługując się teorią liniowych równań różniczkowych otrzymujemy ogólne rozwiązanie równania (3.5)

$$\xi(t) = D\varphi_{x_0,t}(u(\cdot))v(\cdot) = \frac{d}{d\alpha} \Big|_{\alpha=0} \varphi_{x_0,t}(u(\cdot) + \alpha v(\cdot)) = \int_0^t \Phi(t,s)B(s)v(s) ds, \quad (3.6)$$

gdzie $\Phi(t,s)$ jest macierzą fundamentalną układu (3.1) spełniającą równanie $\frac{\partial}{\partial t} \Phi(t,s) = A(t)\Phi(t,s)$ z warunkiem początkowym $\Phi(s,s) = I_n$. Ostatecznie, podstawiając rozwiązanie (3.6) do (3.3) otrzymujemy

$$D\mathcal{F}_{x_0,t}(u(\cdot))v(\cdot) = C(t) \int_0^t \Phi(t,s)B(s)v(s) ds.$$

Dla układu sterowania (3.1) i odwzorowania wejście–wyjście (3.2) sformułujmy zadanie reprodukcji trajektorii wyjścia w następujący sposób. Mając daną trajektorię wyjściową, $y_d(t)$, reprodukowalną w x_0 , znaleźć funkcję sterującą $u_d(t)$, taką że $\mathcal{F}_{x_0}(u_d(\cdot))(t) = y_d(t)$. Przyjmując, że wyjściem są wszystkie zmienne stanu w analogiczny sposób możemy zdefiniować zadanie reprodukcji stanu [55].

Założmy, że dopuszczalne funkcje sterujące są gładkie, klasy C^∞ . W takim razie, możemy znaleźć rozwiązanie zadania reprodukcji trajektorii wnioskując w sposób charakterystyczny dla metody homotopii. Wybieramy gładką krzywą $u_\theta(\cdot) \in \mathcal{U}$, $\theta \in \mathbb{R}$, ze zbioru dopuszczalnych funkcji sterujących przechodzącą przez $u_0(\cdot)$. Niech $y_\theta(t)$ oznacza trajektorię wyjścia układu (3.1) odpowiadającą funkcji sterującej $u_\theta(t)$. Następnie, dla zadanej trajektorii $y_d(t)$ zdefiniujmy funkcję błędu

$$e(t, \theta) = y_\theta(t) - y_d(t) = w(\varphi_{x_0, t}(u_\theta(\cdot))) - y_d(t). \quad (3.7)$$

Naszym zadaniem będzie wyznaczenie sterowania $u_\theta(t)$, dla którego błąd $e(t, \theta)$ będzie zbiegał do zera ze współczynnikiem $\alpha > 0$. Podążając za ideą metody homotopii zdefiniujmy funkcję homotopii jako

$$H(t, \theta) = \frac{\partial e(t, \theta)}{\partial t} + \alpha e(t, \theta) - \left(\frac{\partial e(t, 0)}{\partial t} + \alpha e(t, 0) \right) \exp(-\gamma \theta) \quad (3.8)$$

gdzie współczynnik zbieżności $\gamma > 0$, $t \in [0, T]$, parametr homotopii $\theta \in \mathbb{R}$ i $e(t, 0)$ oznacza błąd dla sterowania początkowego $u(t, 0) = u_0(t)$. Gdy przyrównamy odwzorowanie (3.8) do zera

$$H(t, \theta) = 0$$

wówczas, łatwo wywnioskować, że dla $\theta = 0$ równanie (3.8) jest spełnione automatycznie, a dla θ dążącej do $+\infty$ równanie staje się równaniem błędu

$$\frac{\partial e(t, \theta)}{\partial t} + \alpha e(t, \theta) = 0. \quad (3.9)$$

Żądamy, aby błąd zbiegał eksponencjalnie do zera, co jest spełnione, gdyż rozwiązanie (3.9) jest równe

$$e(t, \theta) = e(t, 0) \exp(-\alpha t).$$

Równanie $H(t, \theta) = 0$ określa rodzinę problemów planowania trajektorii parametryzowanych przez θ . Jeśli możemy rozwiązać problem dla danego θ , wtedy dążąc z parametrem do nieskończoności będziemy w stanie rozwiązać nasz oryginalny problem. Zróżniczkujmy teraz względem θ tożsamość $H(t, \theta) = 0$

$$\frac{\partial^2 e(t, \theta)}{\partial t \partial \theta} + \alpha \frac{\partial e(t, \theta)}{\partial \theta} + \gamma \left(\frac{\partial e(t, 0)}{\partial t} + \alpha e(t, 0) \right) \exp(-\gamma \theta) = 0. \quad (3.10)$$

Korzystając z tego, że

$$\frac{\partial e(t, \theta)}{\partial t} + \alpha e(t, \theta) = \left(\frac{\partial e(t, 0)}{\partial t} + \alpha e(t, 0) \right) \exp(-\gamma \theta),$$

i robiąc odpowiednie podstawienie do (3.10), otrzymujemy następujące równanie błędu

$$\frac{\partial^2 e(t, \theta)}{\partial t \partial \theta} + \alpha \frac{\partial e(t, \theta)}{\partial \theta} + \gamma \frac{\partial e(t, \theta)}{\partial t} + \alpha \gamma e(t, \theta) = 0. \quad (3.11)$$

W celu wyznaczenia równania na sterowanie $u_\theta(t)$, konieczne jest wyznaczenie poniższych elementów posilując się równaniem (3.7)

$$\frac{e(t, \theta)}{\partial t} = C_\theta(t)(f(x_\theta(t)) + G(x_\theta(t))u_\theta(t)) - \dot{y}_d(t),$$

$$\frac{\partial e(t, \theta)}{\partial \theta} = C_\theta(t) \frac{\partial \varphi_{x_0, t}(u_\theta(\cdot))}{\partial \theta} = C_\theta(t) \xi_\theta(t),$$

gdzie macierz $C_\theta(t)$ jest wyznaczana wzdłuż trajektorii $x_\theta(t)$ oraz

$$\xi_\theta(t) = \frac{\partial \varphi_{x_0, t}(u_\theta(\cdot))}{\partial \theta} = D\varphi_{x_0, t}(u_\theta(\cdot)) \frac{du_\theta(\cdot)}{d\theta}.$$

Następnie posługując się (3.5) otrzymujemy

$$\begin{aligned} \frac{\partial^2 e(t, \theta)}{\partial t \partial \theta} &= \dot{C}_\theta(t) \xi_\theta(t) + C_\theta(t) \dot{\xi}_\theta(t) = \\ &\dot{C}_\theta(t) \xi_\theta(t) + C_\theta(t) A_\theta(t) \xi_\theta(t) + C_\theta(t) B_\theta(t) \frac{\partial u(t, \theta)}{\partial \theta}, \end{aligned}$$

gdzie $A_\theta(t)$ i $B_\theta(t)$ muszą być liczone wzdłuż trajektorii $x_\theta(t)$.

Po zastąpieniu odpowiednich elementów w (3.11), dostajemy różniczkowo-całkowe równanie na sterowanie $u(t, \theta)$

$$N_\theta(t) \frac{\partial u(t, \theta)}{\partial \theta} + M_\theta(t) \xi_\theta(t) + \gamma C_\theta(t) (f(x_\theta(t)) + G(x_\theta(t)) u(t, \theta)) = \gamma (\dot{y}_d(t) - \alpha e(t, \theta)), \quad (3.12)$$

gdzie macierze $M_\theta(t) = \dot{C}_\theta(t) + C_\theta(t)(\alpha I_n + A_\theta(t))$, $N_\theta(t) = C_\theta(t) B_\theta(t)$.

Ostatecznie, różniczkując równanie (3.12) względem t i odwołując się do (3.5) otrzymujemy następujące cząstkowe równanie różniczkowe na sterowanie

$$\begin{aligned} N_\theta(t) \frac{\partial^2 u(t, \theta)}{\partial t \partial \theta} + (\dot{N}_\theta(t) + M_\theta(t) B_\theta(t)) \frac{\partial u(t, \theta)}{\partial \theta} + \gamma N_\theta(t) \frac{\partial u(t, \theta)}{\partial t} + \gamma M_\theta(t) B_\theta(t) u(t, \theta) \\ + \dot{M}_\theta(t) \xi_\theta(t) + M_\theta(t) A_\theta(t) \xi_\theta(t) = \gamma (-M_\theta(t) f(x_\theta(t)) + \ddot{y}_d + \alpha \dot{y}_d). \end{aligned} \quad (3.13)$$

Równanie (3.13) będziemy nazywać równaniem reprodukcji trajektorii układu (3.1), które razem z afinicznym układem sterowania i układem wariacyjnym stowarzyszonym z (3.1)

$$\begin{cases} \dot{x}_\theta(t) = f(x_\theta(t)) + G(x_\theta(t)) u_\theta(t) \\ \dot{\xi}_\theta(t) = A_\theta(t) \xi_\theta(t) + B_\theta(t) \frac{\partial u(t, \theta)}{\partial \theta} \end{cases} \quad (3.14)$$

tworzy układ reprodukcji trajektorii [21]. Warunki początkowe i brzegowe dla tego układu są przyjęte jako

$$x_\theta(0) = x_0, \quad \xi_\theta(0) = 0, \quad u(t, 0) = u_0(t),$$

podczas gdy $u(0, \theta)$ jest rozwiązaniem liniowego równania różniczkowego otrzymanego z (3.12)

$$N(0) \frac{du(0, \theta)}{d\theta} + \gamma N(0) u(0, \theta) = \gamma (-C(0) f(x_0) + \dot{y}_d(0) - \alpha (w(x_0) - y_d(0))),$$

gdzie $C(0) = \frac{\partial w(x_0)}{\partial x}$, $N(0) = C(0) G(x_0)$. Układ reprodukcji trajektorii, utworzony przez równania (3.13) i (3.14), będziemy rozwiązywać przy założeniu, że macierz $N_\theta(t)$ jest odwracalna.

Rozdział 4

Przykłady numeryczne — aproksymacja algorytmów kinematyki odwrotnej

W rozdziale zostaną przedstawione wyniki obliczeń numerycznych dla zadania aproksymacji jacobianowych algorytmów kinematyki odwrotnej. Pokażemy przykłady zarówno dla metody wariacyjnej jak i dla metody geometrycznej. Dla metody wariacyjnej rozważymy funkcjonały błędu I i II. Modelami testowymi będą manipulatory i robot mobilny. Wśród wyników symulacji dla manipulatorów przedstawimy przypadki, w których kinematyka robota będzie wyrażona za pomocą współrzędnych [56], a także bez współrzędnych [22], za pomocą specjalnej grupy euklidesowej $SE(3)$. Rozdział kończy porównanie metody opartej na geometrii różniczkowej oraz podejścia wariacyjnego.

W rozdziale pokażemy także, że wszystkie wyprowadzone metody aproksymacji algorytmów jacobianowych można zastosować do rzeczywistego manipulatora przemysłowego. Wybierzemy manipulator POLYCRANK, w którym położenie i orientację efektora określamy w sześciowymiarowej przestrzeni, a jego stopień redundancji wynosi 1.

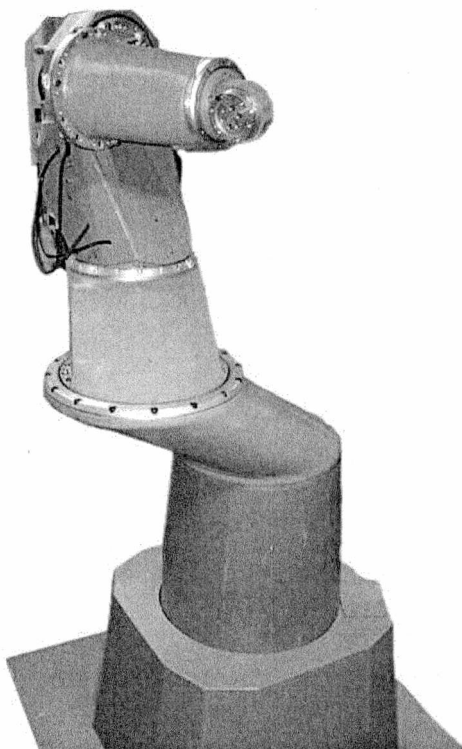
4.1 Manipulator POLYCRANK

Manipulator POLYCRANK, przedstawiony na rysunku 4.1, jest robotem o 7 stopniach swobody skonstruowanym na wydziale Mechanicznym, Energetyki i Lotnictwa Politechniki Warszawskiej. Manipulator ten ma praktycznie nieograniczone zakresy obrotów na przegubach i izotropowe własności kinematyczne i dynamiczne [30, 31].

Parametry Denavita–Hartenberga robota są zawarte w tabeli 4.1. Parametr θ_i oznacza położenie i -tego przegubu, natomiast x_i to położenie i -tego silnika. Parametry geometryczne POLYCRANKa dane są następująco: $l_1 = 0.2975[\text{m}]$, $l_2 = 0.18[\text{m}]$, $l_3 = 1.552[\text{m}]$, $l_4 = l_5 = 0.16[\text{m}]$, $l_6 = 0.2562[\text{m}]$.

Korzystając z parametrów Denavita–Hartenberga, tabela 4.1, otrzymujemy kinematykę manipulatora wyrażoną w $SE(3)$

$$\mathcal{K}(x) = \begin{bmatrix} c_3c_7 - c_6s_3s_7 & -c_6c_7s_3 - c_3s_7 & s_3s_6 & l_1c_1 + l_2c_2 + l_6c_3 + s_3(l_4s_4 + l_5s_5) \\ s_3c_7 + c_6c_3s_7 & c_6c_7c_3 - s_3s_7 & -c_3s_6 & l_1s_1 + l_2s_2 + l_6s_3 - c_3(l_4s_4 + l_5s_5) \\ s_6s_7 & c_7s_6 & c_6 & l_3 + l_4c_4 + l_5c_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4.1)$$



Rysunek 4.1: Manipulator POLYCRANK

Tabela. 4.1: Parametry Denavita–Hartenberga dla manipulatora POLYCRANK

i	θ_i	d_i	a_i	α_i
1	x_1	0	l_1	0
2	$x_2 - x_1$	0	l_2	0
3	$x_3 - x_2 + \frac{\pi}{2}$	l_3	0	$\frac{\pi}{2}$
4	$x_4 + \frac{\pi}{2}$	0	l_4	0
5	$x_5 - x_4$	0	l_5	0
6	$x_6 - x_5 + \frac{\pi}{2}$	l_6	0	$\frac{\pi}{2}$
7	$x_7 + \frac{\pi}{2}$	0	0	0

gdzie $s_i = \sin x_i$ oraz $c_i = \cos x_i$. Odpowiednio, jacobian geometryczny manipulatora przyjmuje postać

$$J_{RT}(x) = \begin{bmatrix} J_R(x) \\ J_T(x) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & c_3 & s_3 s_6 \\ 0 & 0 & 0 & 0 & 0 & s_3 & -c_3 s_6 \\ 0 & 0 & 1 & 0 & 0 & 0 & c_6 \\ -l_1 s_1 & -l_2 s_2 & -l_6 s_3 + c_3(l_4 s_4 + l_5 s_5) & l_4 c_4 s_3 & l_5 c_5 s_3 & 0 & 0 \\ l_1 c_1 & l_2 c_2 & l_6 c_3 + s_3(l_4 s_4 + l_5 s_5) & -l_4 c_3 c_4 & -l_5 c_3 c_5 & 0 & 0 \\ 0 & 0 & 0 & -l_4 s_4 & -l_5 s_5 & 0 & 0 \end{bmatrix}.$$

Niech kinematyka wyrażona za pomocą współrzędnych określa położenie końca efektora w przestrzeni kartezjańskiej i jego orientację wyrażoną przez kąty ZXZ Eulera

$$k(x) = \begin{pmatrix} l_1 \cos x_1 + l_2 \cos x_2 + l_6 \cos x_3 + \sin x_3 (l_4 \sin x_4 + l_5 \sin x_5) \\ l_1 \sin x_1 + l_2 \sin x_2 + l_6 \sin x_3 - \cos x_3 (l_4 \sin x_4 + l_5 \sin x_5) \\ l_3 + l_4 \cos x_4 + l_5 \cos x_5 \\ x_3 \\ x_6 \\ x_7 \end{pmatrix}. \quad (4.2)$$

Obliczmy jacobian manipulatora

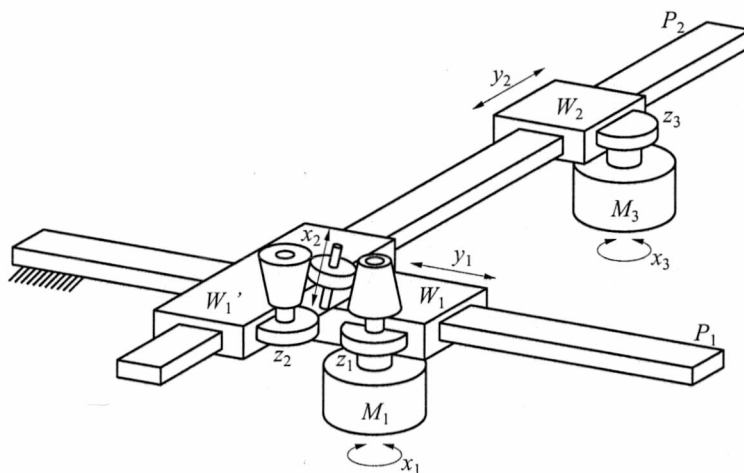
$$J(x) = \frac{\partial k(x)}{\partial x} = \begin{bmatrix} -l_1 s_1 & -l_2 s_2 & (l_4 s_4 + l_5 s_5) c_3 - l_6 s_3 & l_4 s_3 c_4 & l_5 s_3 c_5 & 0 & 0 \\ l_1 c_1 & l_2 c_2 & (l_4 s_4 + l_5 s_5) s_3 + l_6 c_3 & -l_4 c_3 c_4 & -l_5 c_3 c_5 & 0 & 0 \\ 0 & 0 & 0 & -l_4 s_4 & -l_5 s_5 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Manipulator POLYCRANK znajduje się w konfiguracjach osobliwych dla wartości zmiennych przegubowych równych $x_5 = j\pi$ lub $x_2 - x_1 = l\pi$, $j, l = 0, \pm 1, \dots$

4.2 Metoda wariacyjna — Funkcjonał błędu I dla kinematyki we współrzędnych

4.2.1 Manipulator 3R bez konfiguracji osobliwych

Manipulator jest planarnym manipulatorem o 3 stopniach swobody [53], przedstawionym na rysunku 4.2, skonstruowanym w Zakładzie Podstaw Cybernetyki i Robotyki Politechniki Wrocławskiej, według projektu dra inż. K. Mianowskiego z Politechniki Warszawskiej, twórcy robota POLYCRANK.



Rysunek 4.2: Manipulator

Niech zmienne przegubowe określone będą przez (x_1, x_2, x_3) , a dwie zmienne zadaniowe (y_1, y_2) określają pozycję wózka W_2 w układzie kartezjańskim związanym z podstawą. Zmienna przegubowa x_1 jest poruszana bezpośrednio za pomocą silnika M_1 , podobnie zmienna x_3 jest zależna wyłącznie od silnika M_3 . Pozycja W_2 wzdłuż prowadnicy P_2 zależy od współrzędnej x_3 oraz kąta obrotu koła zębatego z_2 . Kąt ten jest zadawany poprzez przekładnię, której przełożenie jest modyfikowane zmienną x_2 . Kinematyka manipulatora jest opisana przez równania $y_1 = c_1 x_1$ oraz $y_2 = f_2(x_2) + c_3 x_3$, gdzie c_1 i c_3 są stałymi a funkcja $f_2(x_2) = c_2 \frac{r_1 + a x_2}{r_2 - a x_2}$, dla parametru $a = \frac{r_2 - r_1}{l}$; r_1, r_2 i l są odpowiednio promieniami i długością tworzącej stożka ściętego będącego elementem przekładni. Zmieniając współrzędne $q_1 = c_1 x_1, q_2 = c_3 x_3, q_3 = \frac{1}{c_1} f_2(x_2)$ i powracając do pierwotnej notacji $q = x$, otrzymujemy

$$k(x) = (x_1, x_2 + x_1 x_3).$$

Po dokonaniu stosownych obliczeń obliczamy jacobian

$$J(x) = \frac{\partial k(x)}{\partial x} = \begin{bmatrix} 1 & 0 & 0 \\ x_3 & 1 & x_1 \end{bmatrix},$$

macierz manipulowalności

$$M(x) = J(x)J^T(x) = \begin{bmatrix} 1 & x_3 \\ x_3 & 1 + x_1^2 + x_3^2 \end{bmatrix},$$

generator jądra jacobianu

$$K(x) = \frac{1}{\sqrt{1 + x_1^2}} (0 \quad -x_1 \quad 1)^T$$

oraz manipulowalność $m(x) = \sqrt{1 + x_1^2}$.

Naszym zadaniem jest znalezienie takiej funkcji rozszerzającej $h(x)$, która będzie minimalizowała funkcjonal błędu I. Dla przykładu porównamy rozwiązanie otrzymane przy użyciu metody Maciejewskiego i Roberta, z równaniem błędu (2.21), z funkcjonalem błędu I zadany równaniem (2.29).

Metoda Roberta i Maciejewskiego

Dla uproszczenia obliczeń założymy, że $h(x) = x_3 f(x_1)$, $f(x_1) \neq 0$. Rozwiązanie zadania aproksymacji prowadzi do następującego równania Eulera-Lagrange'a dla funkcjonału (2.21)

$$2x_3 \left(f^2(x_1) + 4(1 + x_1^2) \left(\frac{df(x_1)}{dx_1} \right)^2 - f(x_1) \left(5x_1 \frac{df(x_1)}{dx_1} + (1 + x_1^2) \frac{d^2 f(x_1)}{dx_1^2} \right) \right) = 0.$$

Analityczne rozwiązanie powyższego równania możemy otrzymać korzystając z programu MATHEMATICA

$$f(x_1) = \frac{C_1 \sqrt{1 + x_1^2}}{\left(x_1 \sqrt{1 + x_1^2} + \sinh^{-1}(x_1) + C_2 \right)^{1/3}}, \quad (4.3)$$

gdzie C_1 i C_2 oznaczają stałe całkowania. Optymalna funkcja rozszerzająca przybiera postać [56]

$$h(x) = x_3 \frac{C_1 \sqrt{1 + x_1^2}}{\left(x_1 \sqrt{1 + x_1^2} + \sinh^{-1}(x_1) + C_2 \right)^{1/3}}. \quad (4.4)$$

Na potrzeby późniejszego porównania rozwińmy mianownik (4.3) w szereg Taylora wokół $x_1 = 0$. Po podstawieniu $C_1 = C_2 = 1$ otrzymujemy

$$f(x_1) \cong \left(1 - \frac{2}{3}x_1\right) \sqrt{1 + x_1^2},$$

a co za tym idzie, aproksymująca funkcja rozszerzająca jest równa

$$h(x_1, x_3) \cong x_3 \left(1 - \frac{2}{3}x_1\right) \sqrt{1 + x_1^2}. \quad (4.5)$$

Funkcjonał błędu I

Teraz naszym celem jest wyznaczenie funkcji rozszerzającej $h(x)$ minimalizującej funkcjonal błędu (2.29). Zadanie rozwiązemy, tak jak poprzednio, odwołując się do równania Eulera–Lagrange’a. Aby móc to zrobić, powinniśmy najpierw wyliczyć

$$P(x) = \frac{1}{1 + x_1^2} \begin{bmatrix} 1 + x_1^2 & -x_3 & -x_1x_3 \\ -x_3 & \frac{1+x_1^2+x_3^2+x_1^4}{1+x_1^2} & \frac{x_1x_3^2-x_1^3}{1+x_1^2} \\ -x_1x_3 & \frac{x_1x_3^2-x_1^3}{1+x_1^2} & \frac{1+2x_1^2+x_1^2x_3^2}{1+x_1^2} \end{bmatrix}.$$

Po dokonaniu odpowiednich podstawień, równanie Eulera–Lagrange’a (2.49) przybiera następującą postać

$$\begin{aligned} & \sqrt{1 + x_1^2} \frac{\partial^2 h(x)}{\partial x_1^2} + \frac{1 + x_1^2 + x_3^2 + x_1^4}{(1 + x_1^2)^{3/2}} \frac{\partial^2 h(x)}{\partial x_2^2} \\ & + \frac{1 + 2x_1^2 + x_1^2x_3^2}{(1 + x_1^2)^{3/2}} \frac{\partial^2 h(x)}{\partial x_3^2} - \frac{2x_3}{(1 + x_1^2)^{1/2}} \frac{\partial^2 h(x)}{\partial x_1 \partial x_2} - \frac{2x_1x_3}{(1 + x_1^2)^{1/2}} \frac{\partial^2 h(x)}{\partial x_1 \partial x_3} \\ & + \frac{2x_1(x_3^2 - x_1^2)}{(1 + x_1^2)^{3/2}} \frac{\partial^2 h(x)}{\partial x_2 \partial x_3} + \frac{3x_1x_3}{(1 + x_1^2)^{3/2}} \frac{\partial h(x)}{\partial x_2} + \frac{2x_1^2x_3 - x_3}{(1 + x_1^2)^{3/2}} \frac{\partial h(x)}{\partial x_3} = 0. \end{aligned} \quad (4.6)$$

Założmy warunki brzegowe $h(x_1, x_2, 0) = 0$ i, tak jak w poprzednim przykładzie, $h(x) = x_3 f(x_1)$. Przy takich założeniach równanie (4.6) redukuje się do równania różniczkowego zwyczajnego drugiego rzędu

$$\frac{d^2 f(x_1)}{dx_1^2} - \frac{2x_3}{1 + x_1^2} \frac{df(x_1)}{dx_1} + \frac{2x_1^2 - 1}{(1 + x_1^2)^2} f(x_1) = 0. \quad (4.7)$$

Podobnie jak wyżej, korzystając z programu MATHEMATICA otrzymujemy rozwiązanie analityczne (4.7)

$$f(x_1) = (ax_1 + b) \sqrt{1 + x_1^2},$$

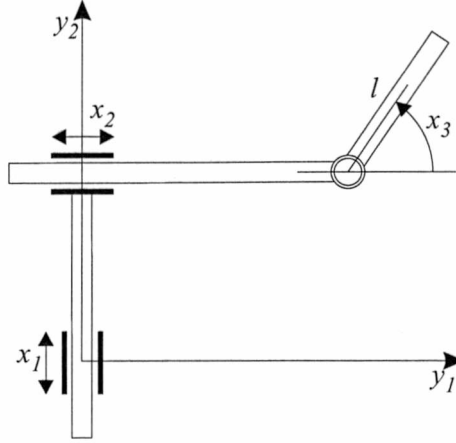
gdzie a i b są stałymi całkowania [56]. W rezultacie, optymalna funkcja rozszerzająca jest równa

$$h_{a,b}(x) = x_3(ax_1 + b) \sqrt{1 + x_1^2}. \quad (4.8)$$

Porównując rozwiązania (4.5) i (4.8) można zauważyć, że pomimo różnych postaci funkcjonalów błędu, otrzymane funkcje rozszerzające są sobie równe dla $a = 1$ i $b = -\frac{2}{3}$.

4.2.2 Manipulator TTR

Kolejnym przykładem jest planarny manipulator o 3 stopniach swobody przedstawiony na rysunku 4.3. Jego kinematyka jest opisana równaniem



Rysunek 4.3: Manipulator TTR

$$k(x) = (x_2 + l \cos x_3, x_1 + l \sin x_3).$$

Jakobian manipulatora, macierz manipulowalności, generator jądra jakobianu oraz funkcja manipulowalności są odpowiednio równe

$$J(x) = \begin{bmatrix} 0 & 1 & -l \sin x_3 \\ 1 & 0 & l \cos x_3 \end{bmatrix}, \quad M(x) = \begin{bmatrix} l + l^2 \sin^2 x_3 & -l^2 \sin x_3 \cos x_3 \\ -l^2 \sin x_3 \cos x_3 & l + l^2 \cos^2 x_3 \end{bmatrix},$$

$$K(x) = \frac{1}{\sqrt{1+l^2}} (-l \cos x_3, l \sin x_3, 1)^T, \quad m(x) = \sqrt{1+l^2}.$$

Ponownie, naszym zadaniem jest znalezienie jednej funkcji rozszerzającej $h(x)$, która będzie minimalizowała błąd aproksymacji (2.29). Do tego celu wykorzystamy również równanie Eulera–Lagrange’a. Korzystając z równania (2.49), po stosownych obliczeniach nadajemy temu równaniu postać

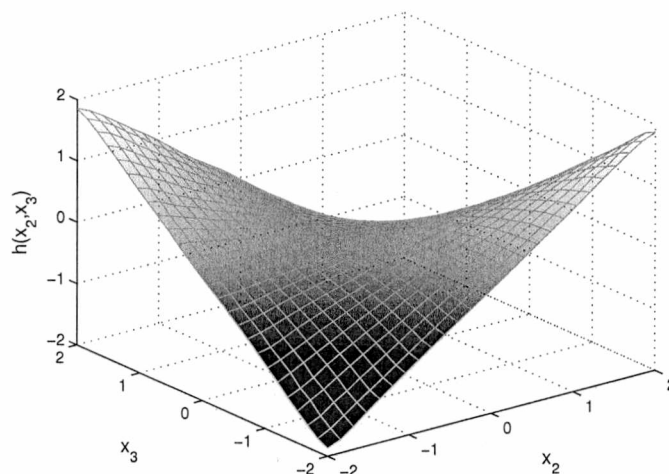
$$\text{tr} \frac{\partial}{\partial x} \left(R(x) \frac{\partial h(x)}{\partial x} \right) = 0,$$

gdzie $R(x) = m(x)P(x)$. Niech długość ramienia $l = 1$, wtedy współczynniki macierzy $R(x)$ są równe

$$R(x) = \frac{\sqrt{2}}{2} \begin{bmatrix} \frac{7-\cos(2x_3)}{4} & \frac{\cos x_3 \sin x_3}{2} & -\frac{\cos x_3}{2} \\ \frac{\cos x_3 \sin x_3}{2} & \frac{7+\cos(2x_3)}{4} & \frac{\sin x_3}{2} \\ -\frac{\cos x_3}{2} & \frac{\sin x_3}{2} & \frac{3}{2} \end{bmatrix}.$$

Ze względu na trudności obliczeniowe, załóżmy arbitralnie, że $h(x) = h(x_2, x_3)$. W tym wypadku macierz $R(x)$ redukuje się do

$$R(x) = \frac{\sqrt{2}}{2} \begin{bmatrix} \frac{7+\cos(2x_3)}{4} & \frac{\sin x_3}{2} \\ \frac{\sin x_3}{2} & \frac{3}{2} \end{bmatrix}.$$

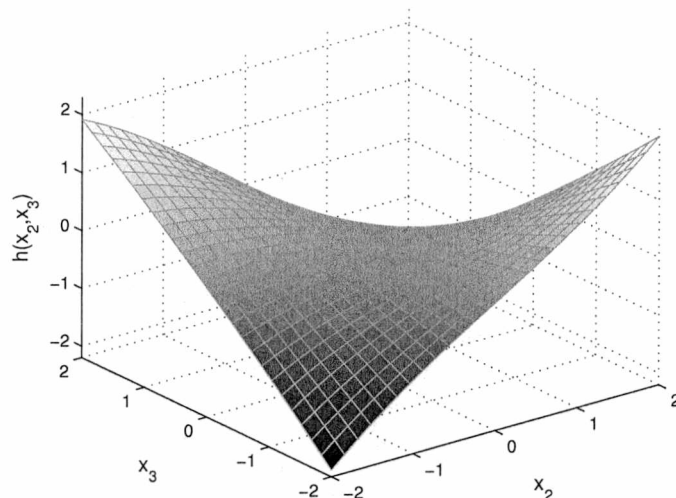


Rysunek 4.4: Optymalna funkcja rozszerzająca

Numeryczne rozwiązanie zadania przy warunkach brzegowych $h(x_2, -2) = x_2$, $h(-2, x_3) = x_3$, $h(x_2, 2) = -x_2$, $h(2, x_3) = -x_3$, otrzymane przy pomocy pakietu PDE toolbox programu MATLAB jest przedstawione na rysunku 4.4. Na potrzeby dalszych obliczeń, aproksymujemy otrzymane rozwiązanie wielomianem trzeciego stopnia

$$\begin{aligned}
 h(x_2, x_3) = & 3.5760 \cdot 10^{-2} x_2^3 - 8.2205 \cdot 10^{-6} x_3^3 + 1.6717 \cdot 10^{-5} x_2^2 x_3 + 1.5522 \cdot 10^{-2} x_2 x_3^2 \\
 & + 1.4597 \cdot 10^{-2} x_2^2 - 4.0394 \cdot 10^{-5} x_3^2 - 5.0074 \cdot 10^{-1} x_2 x_3 - 1.5981 \cdot 10^{-1} x_2 \\
 & + 7.4522 \cdot 10^{-6} x_3 + 2.8858 \cdot 10^{-5},
 \end{aligned}$$

które jest przedstawione na rysunku 4.5.



Rysunek 4.5: Aproksymowana funkcja rozszerzająca

Mając analityczną postać funkcji rozszerzającej, możemy wyliczyć jacobian rozszerzony

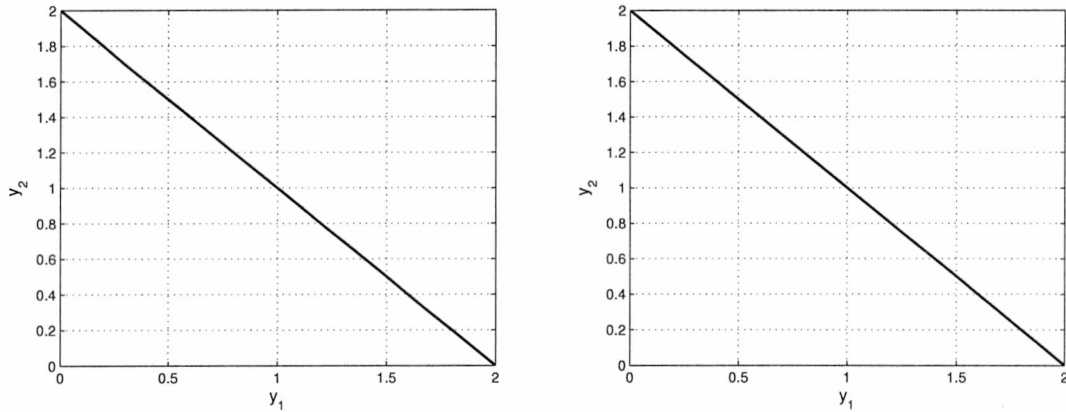
$$\bar{J}(x) = \begin{bmatrix} 0 & 1 & -\sin x_3 \\ 1 & 0 & \cos x_3 \\ 0 & j_{32}(x) & j_{33}(x) \end{bmatrix},$$

gdzie

$$\begin{aligned} j_{32}(x) &= 1.0728 \cdot 10^{-1} x_2^2 + 3.3434 \cdot 10^{-5} x_2 x_3 + 1.5522 \cdot 10^{-2} x_3^2 + 2.9194 \cdot 10^{-5} x_2 \\ &\quad - 5.0074 \cdot 10^{-1} x_3 - 1.5981 \cdot 10^{-1}, \\ j_{33}(x) &= -2.4660 \cdot 10^{-5} x_3^2 + 1.6717 \cdot 10^{-5} x_2^2 + 3.1044 \cdot 10^{-2} x_2 x_3 - 8.0788 \cdot 10^{-5} x_3 \\ &\quad - 5.0074 \cdot 10^{-1} x_2 + 7.4522 \cdot 10^{-6} \end{aligned}$$

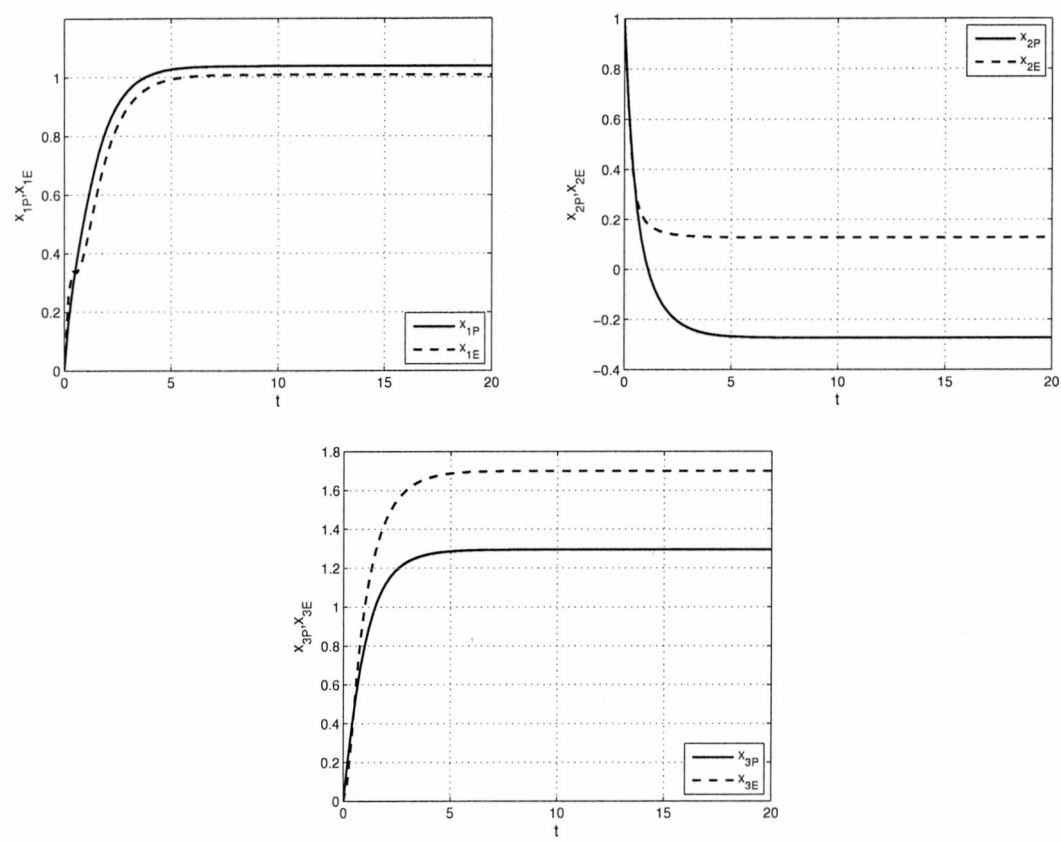
i odpowiednią odwrotność typu jacobianu rozszerzonego.

Dla tak przygotowanych danych porównajmy rozwiązania zadania odwrotnej kinematyki otrzymane za pomocą algorytmu typu jacobianu pseudoodwrotnego i typu jacobianu rozszerzonego. Zadanie polega na osiągnięciu przez koniec efektora manipulatora punktu $y = (0, 2)$ w przestrzeni zadaniowej. Konfiguracja początkowa $x_0 = (0, 1, 0)$, co odpowiada położeniu $y_0 = (2, 0)$ w przestrzeni zadaniowej. Rysunek 4.6 przedstawia ścieżki końca efektora dla algorytmu typu jacobianu pseudoodwrotnego i dla algorytmu typu jacobianu rozszerzonego. Porównanie trajektorii współrzędnych przegubowych jest zobrazowane na rysunku 4.7. Łatwo zauważyć, że oba algorytmy rozwiązały z powodzeniem zadanie kinematyki odwrotnej, a wygenerowane trajektorie przegubowe są bardzo bliskie siebie. Można jednak oczekiwać, że dla niektórych zadań różnica w otrzymywanych trajektoriach może być dużo większa, ponieważ otrzymana funkcja rozszerzająca jest optymalna w sensie średnim na pewnym, założonym obszarze.



Rysunek 4.6: Ścieżka w przestrzeni zadaniowej otrzymana za pomocą algorytmu jacobianu pseudoodwrotnego (z lewej) i jacobianu rozszerzonego (z prawej)

Kolejna symulacja będzie miała na celu sprawdzenie powtarzalności algorytmu. Wybierzmy cztery punkty w przestrzeni zadaniowej $A(0, 1)$, $B(0, 2)$, $C(1, 2)$ i $D(1, 1)$. Zadanie będzie polegało na ruchu efektora pomiędzy tymi czterema punktami tworząc krzywą zamkniętą. Innymi słowy mówiąc, zaczynamy ruch od punktu A i wracamy do niego przechodząc przez punkty B , C oraz D . W każdym kroku punktem początkowym jest rozwiązanie otrzymane z poprzedniego kroku. Wyniki symulacji dla jacobianu pseudoodwrotnego i jacobianu rozszerzonego zawarte są w tabeli 4.2. Jak widać, algorytm typu jacobianu rozszerzonego jest powtarzalny, w przeciwieństwie do algorytmu typu jacobianu pseudoodwrotnego.



Rysunek 4.7: Trajektorie przegubowe

Tabela. 4.2: Sprawdzenie powtarzalności

Algorytm		Jakobian rozszerzony		
A	x_0	0	0	1.5708
B	x_{1f}	0	0	1.5708
C	x_{2f}	1.9633	0.0007	0.0367
D	x_{3f}	0.9633	0.0007	0.0367
A	x_{4f}	0	0	1.5708

Algorytm		Jakobian pseudoodwrotny		
A	x_0	0	0	1.5708
B	x_{1f}	1	0	1.5708
C	x_{2f}	1.1132	0.5379	1.0904
D	x_{3f}	0.2804	0.3056	0.8032
A	x_{4f}	0.0603	-0.3420	1.2217

4.2.3 Manipulator POLYCRANK

Badania symulacyjne przeprowadzono dla kinematyki manipulatora POLYCRANK wyrażonej we współrzędnych formułą (4.2). Ponieważ stopień redundancji robota $s = 1$, naszym zadaniem jest znalezienie jednej funkcji rozszerzającej $h(x)$, która będzie minimalizowała funkcjonal błędu $\mathcal{E}_1(h)$ określony równaniem (2.29). Zadanie rozwiążemy metodą Ritza. Założmy, że funkcja rozszerzająca jest funkcją liniową $h(x) = a^\top x$, $a \in \mathbb{R}^7$, natomiast aproksymacja będzie przeprowadzona na obszarze

$$\mathbb{Q} = \left\{ x \in \mathbb{R}^7 \mid 0.001 \leq x_1 \leq \frac{\pi}{2}, \frac{\pi}{2} + 0.001 \leq x_2 \leq \pi, 0 \leq x_3 \leq \pi, 0.001 \leq x_4 \leq \frac{\pi}{2}, \right. \\ \left. \frac{\pi}{2} + 0.001 \leq x_5 \leq \pi - 0.001, 0.001 \leq x_6 \leq \frac{\pi}{2}, 0 \leq x_7 \leq \pi \right\}. \quad (4.9)$$

Optymalna funkcja rozszerzająca jest równa

$$h(x) = -0.016516x_1 - 0.019845x_2 + 0.023433x_3 - 0.037846x_4 + 0.037855x_5, \quad (4.10)$$

a odpowiednia wartość funkcjonału błędu $\mathcal{E}_1(h) = 0.763599$. Za pomocą funkcji rozszerzającej (4.10) możemy zdefiniować odpowiedni jacobian rozszerzony i zdefiniować algorytm typu jacobianu rozszerzonego. Obliczenia przeprowadźmy dla trzech różnych zadań. We wszystkich symulacjach $\gamma = 3$ a czas symulacji $t_{\max} = 5$. W tabeli 4.3 zostały zebrane konfiguracje początkowe i punkty zadane w przestrzeni zadaniowej dla odpowiednich zadań. Na rysunkach 4.8, 4.9, 4.10 zostały przedstawione rozwiązania zadań. Przebiegi normy

Tabela. 4.3: Dane potrzebne do obliczeń

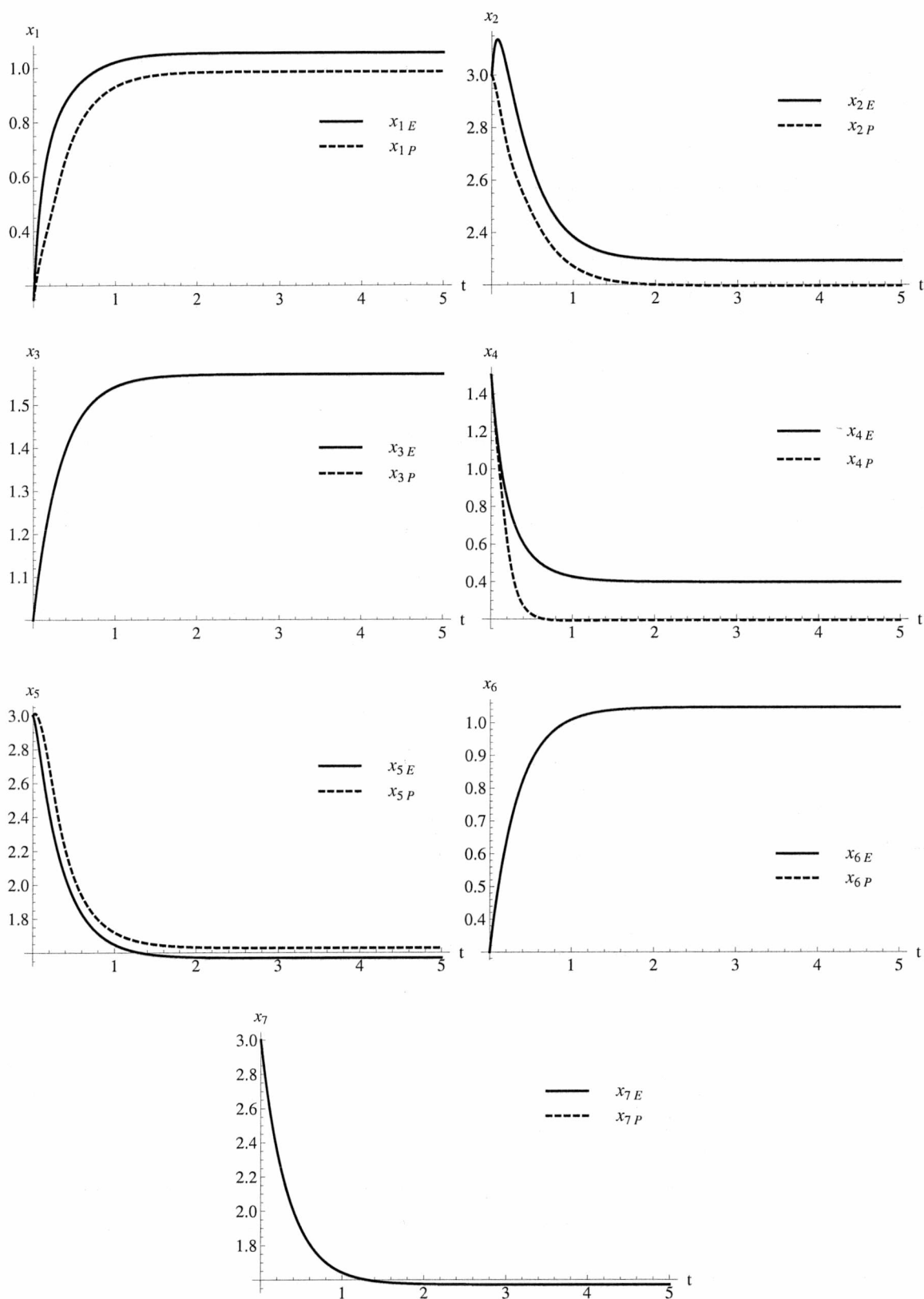
	Konfiguracja początkowa	Punkt końcowy
Zadanie 1	$x_0 = (0.15, 3, 1, 1.5, 3, 0.3, 3)$	$y_d = (0.25, 0.65, 1.7, \frac{\pi}{2}, \frac{\pi}{3}, \frac{\pi}{2})$
Zadanie 2	$x_0 = (1, 1.7, 0.2, 1.5, 3, 0.5, 3)$	$y_d = (0.21, 0.61, 1.7, \frac{\pi}{2}, \frac{\pi}{3}, \frac{\pi}{3})$
Zadanie 3	$x_0 = (1.5, 2, 3, 0.4, 2, 0.5, 0.5)$	$y_d = (0.21, 0.51, 1.5, \frac{\pi}{2}, \frac{\pi}{3}, \frac{\pi}{3})$

prędkości przegubowych w czasie pokazane są na rysunkach 4.11, 4.12 i 4.13, gdzie, dla porównania, wykreślono również przebiegi normy prędkości dla pewnej wybranej losowo, nieoptymalnej funkcji rozszerzającej. W tabeli 4.4 zebrano wartości kryteriów

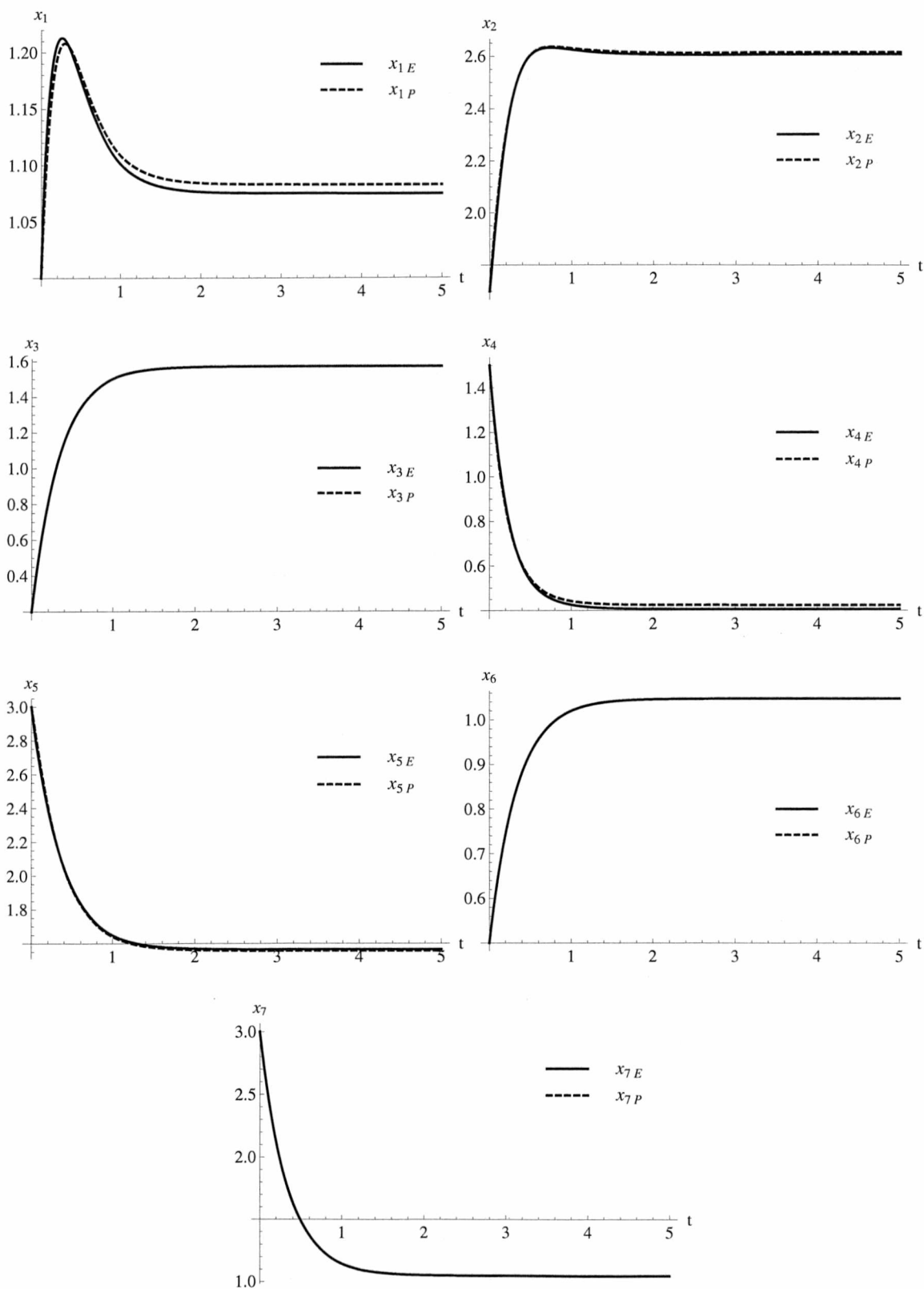
$$\delta_1 = \frac{\sqrt{\int_{t_0}^{t_{\max}} \|x_P(t) - x_E(t)\|^2 dt}}{\sqrt{\int_{t_0}^{t_{\max}} \|x_P(t)\|^2 dt}}, \quad (4.11)$$

$$\delta_2 = \frac{\sqrt{\int_{t_0}^{t_{\max}} \|\dot{x}_P(t) - \dot{x}_E(t)\|^2 dt}}{\sqrt{\int_{t_0}^{t_{\max}} \|\dot{x}_P(t)\|^2 dt}}, \quad (4.12)$$

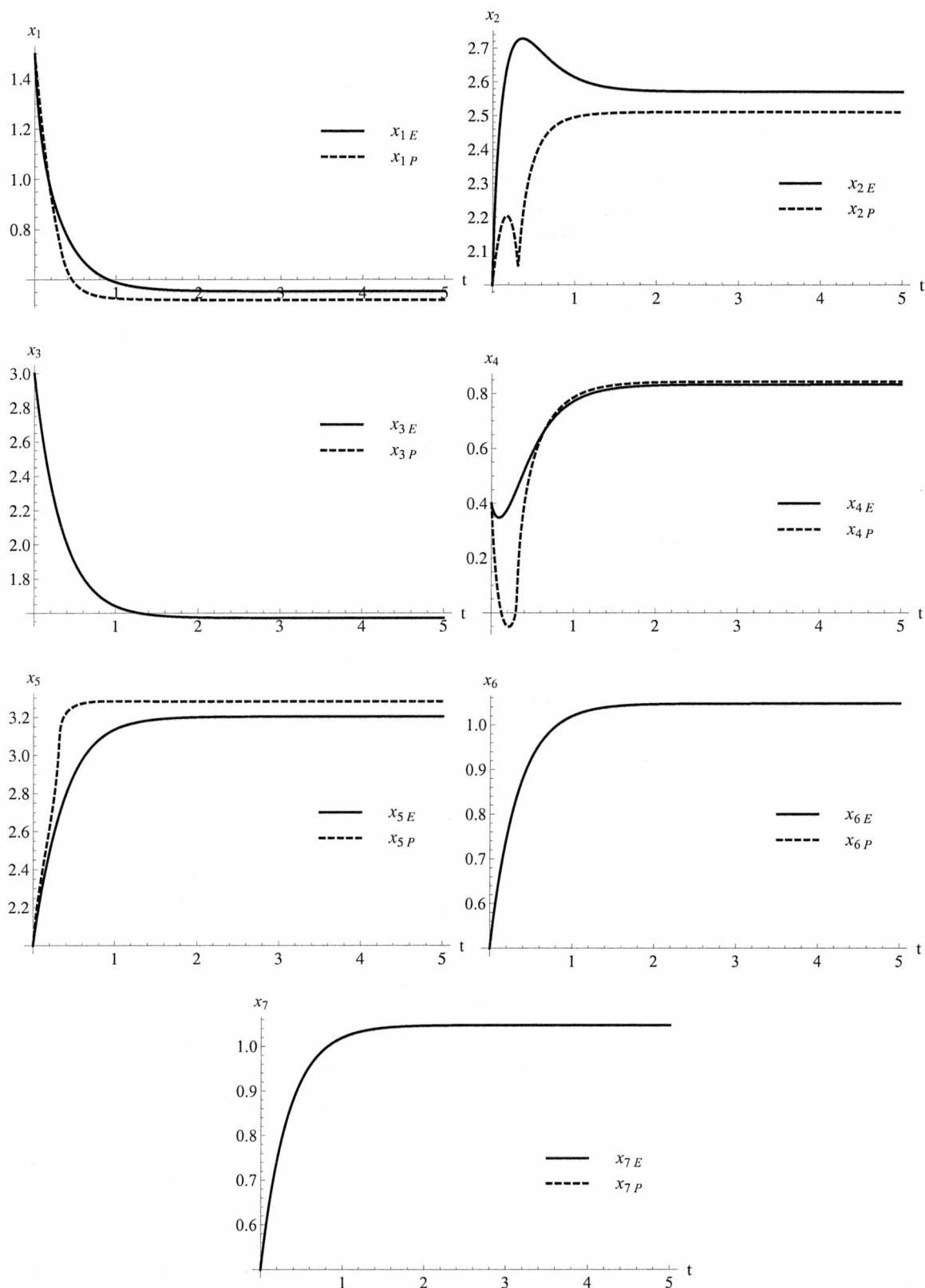
będących miarą jakości aproksymacji. Odpowiadają one średniej, względnej różnicy trajektorii oraz średniej, względnej różnicy prędkości w przegubach otrzymanych z obu algorytmów na całym przedziale czasu symulacji.



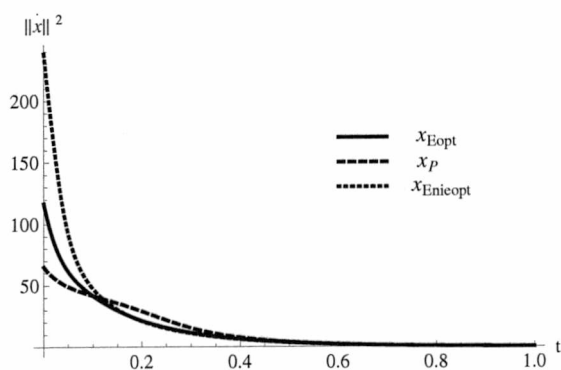
Rysunek 4.8: Rozwiązanie zadania 1



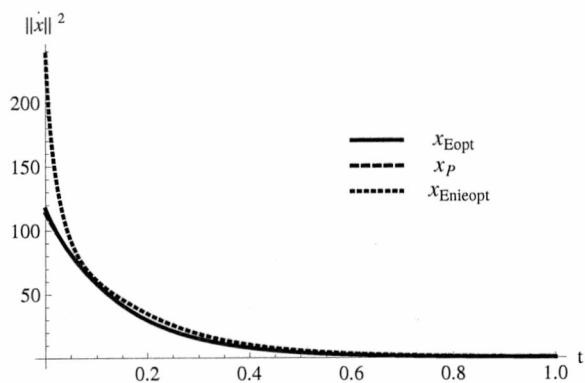
Rysunek 4.9: Rozwiązanie zadania 2



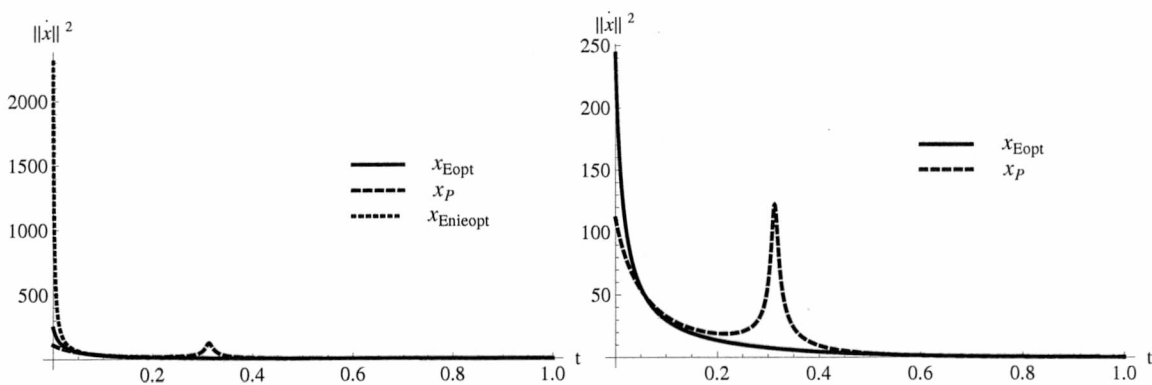
Rysunek 4.10: Rozwiązanie zadania 3



Rysunek 4.11: Normy prędkości przegubowych dla zadania 1



Rysunek 4.12: Normy prędkości przegubowych dla zadania 2



Rysunek 4.13: Normy prędkości przegubowych dla zadania 3

Tabela. 4.4: Wartości miar jakości aproksymacji dla zadań 1, 2 i 3

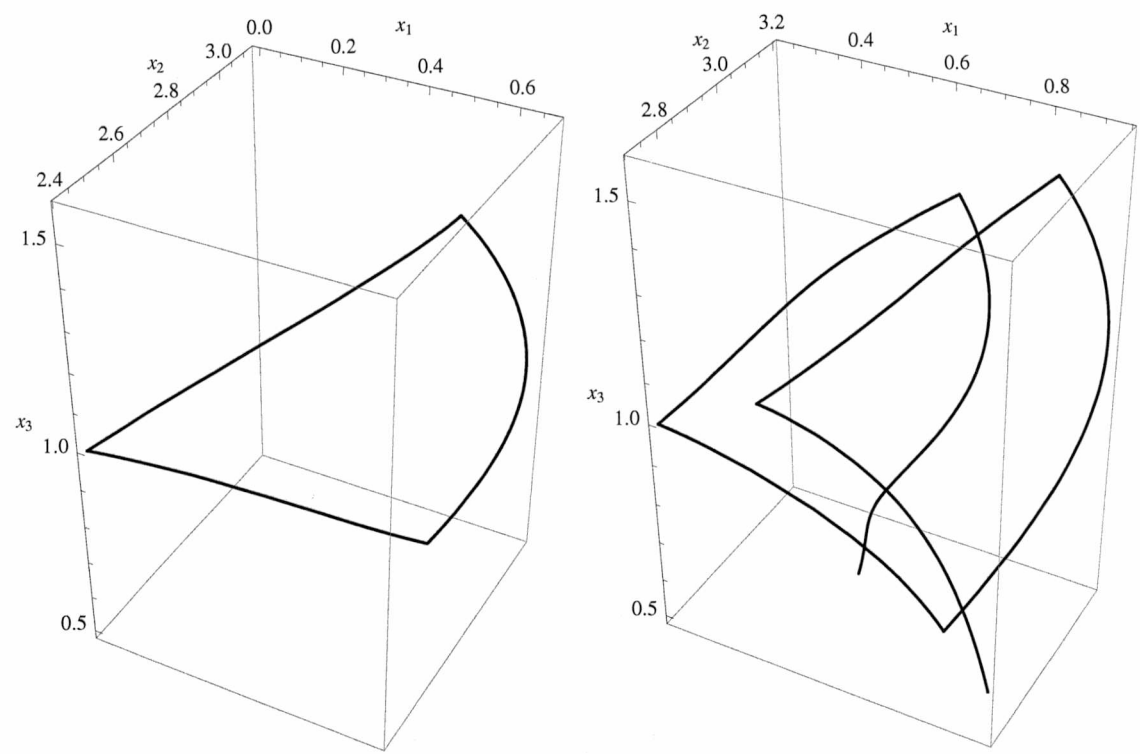
	Jakobian rozszerzony optymalny		Jakobian rozszerzony nieoptymalny	
	δ_1	δ_2	δ_1	δ_2
Zadanie 1	0.07235	0.44217	0.23003	0.80258
Zadanie 2	0.00566	0.06068	0.30564	0.53602
Zadanie 3	0.05162	0.67504	0.11714	1.07615

Analizując rysunki dochodzimy do wniosku, że zadanie aproksymacji zostało pomyślnie rozwiązane. Wynikowy algorytm typu jakobianu rozszerzonego jest rozsądną aproksymacją algorytmu typu jakobianu pseudoodwrotnego. Ponieważ aproksymacja odbywa się na pewnym obszarze, otrzymana funkcja rozszerzająca, a co za tym idzie jakobian rozszerzony, jest optymalna w sensie średnim. Jak widać na rysunku 4.11, generowane średnie wartości prędkości przegubowych w trakcie ruchu są większe od tych otrzymanych z algorytmu jakobianu typu pseudoodwrotnego, nie mniej jednak lepsze od wyniku uzyskanego dla losowo wybranej funkcji rozszerzającej. Fakt, że w niektórych przypadkach (por. rysunek 4.13) norma prędkości przegubowych dla jakobianu rozszerzonego jest mniejsza od otrzymanej za pomocą jakobianu pseudoodwrotnego, wynika z tego, że ten ostatni dostarcza rozwiązanie będące minimum lokalnym. Zdarzają się także przypadki, w których rozwiązanie algorytmu typu jakobianu rozszerzonego może być bardzo podobne lub być wręcz identyczne z rozwiązaniem algorytmu typu jakobianu pseudoodwrotnego, rysunek 4.9.

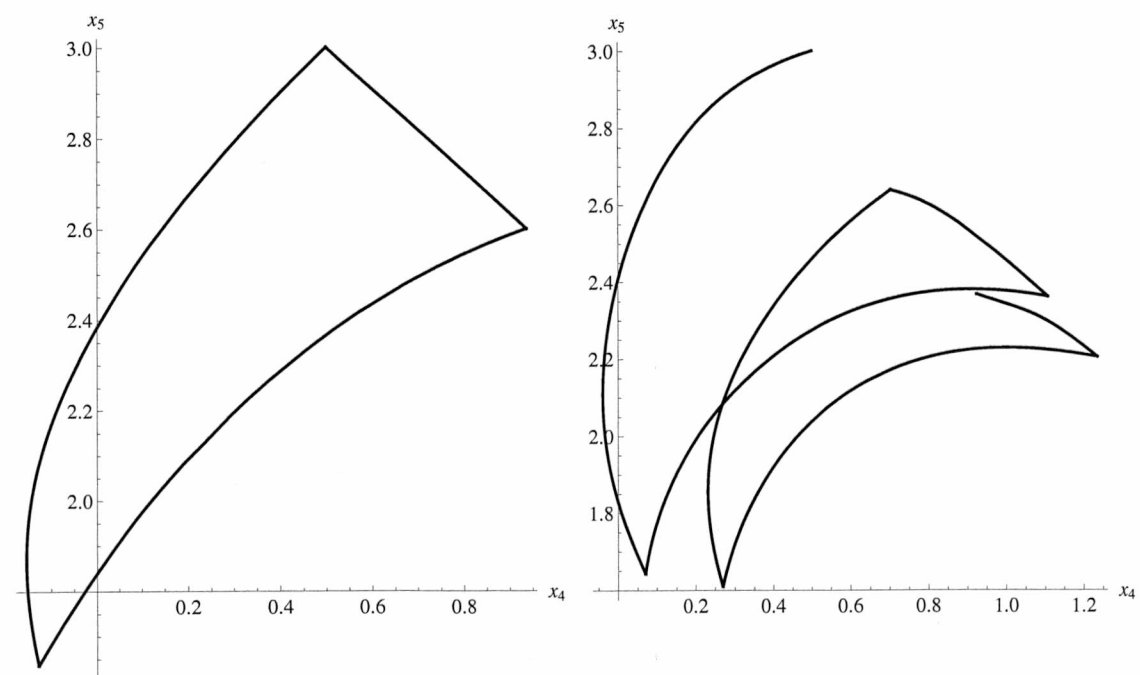
Aby zbadać własność powtarzalności otrzymanych algorytmów, przeprowadzono następujące doświadczenie. Wybrano trzy punkty w przestrzeni zadaniowej robota $A = (0.35532, 0.203727, 1.53401, 0.5, 0.5, 3)$, $B = (0.21, 0.51, 1.7, \frac{\pi}{2}, \frac{\pi}{3}, \frac{\pi}{3})$ oraz punkt $C = (0.48, 0.22, 1.51, 1, 1, 2)$. Rozwiązywanym zadaniem jest przeprowadzenie końca efektora manipulatora z punktu A pomiędzy punktami B, C z powrotem do punktu A , tworząc krzywą zamkniętą w przestrzeni zadaniowej. Dla przejrzystości wyników, zakreślamy tę krzywą dwa razy. Wyniki obliczeń zawarte są w tabeli 4.5. Na ich podstawie można wyciągnąć wniosek, że algorytm kinematyki odwrotnej oparty na jakobianie rozszerzonym posiada własność powtarzalności w przeciwieństwie do algorytmu typu jakobianu pseudoodwrotnego. Dodatkowo, rysunki 4.14–4.16 przedstawiają graficzne potwierdzenie tej tezy.

Tabela. 4.5: Sprawdzenie powtarzalności

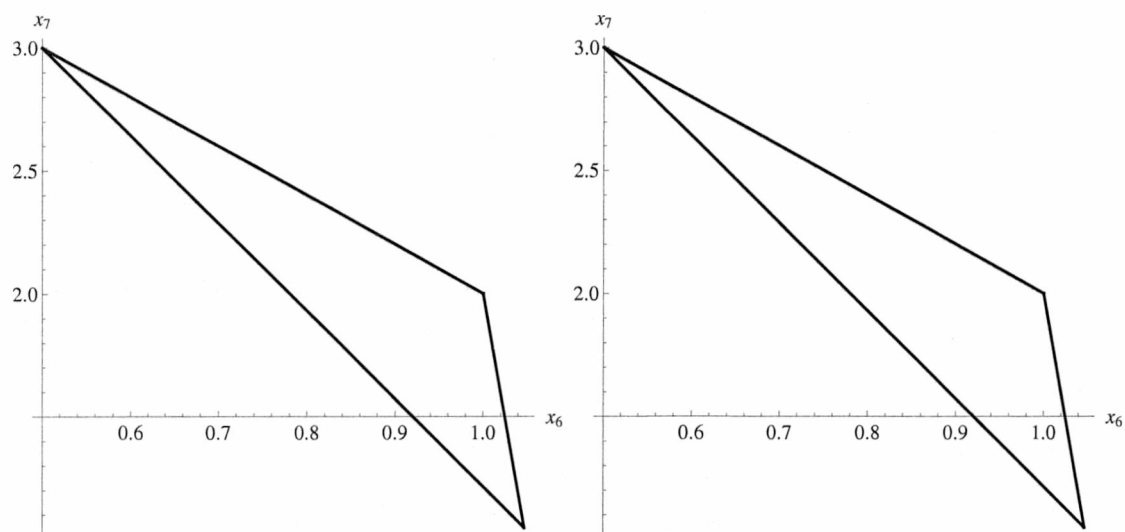
Algorytm		Jakobian rozszerzony						
A	x_0	0.5	3	0.5	0.5	3	0.5	3
B	x_{1f}	0.657333	2.72992	1.5708	-0.126507	1.63786	1.0472	1.0472
C	x_{2f}	-0.0045703	2.41081	1	0.935985	2.59736	1	2
A	x_{3f}	0.5	3	0.5	0.5	3	0.5	3
B	x_{4f}	0.657333	2.72992	1.5708	-0.126507	1.63786	1.0472	1.0472
C	x_{5f}	-0.0045703	2.41081	1	0.935985	2.59736	1	2
A	x_{6f}	0.5	3	0.5	0.5	3	0.5	3
Algorytm		Jakobian pseudoodwrotny						
A	x_0	0.5	3	0.5	0.5	3	0.5	3
B	x_{1f}	0.771934	2.88152	1.5708	0.0700009	1.64341	1.0472	1.0472
C	x_{2f}	0.226379	2.70766	1	1.10556	2.3619	1	2
A	x_{3f}	0.739445	2.92491	0.5	0.701542	2.63884	0.5	3
B	x_{4f}	0.894524	3.02032	1.5708	0.269479	1.60972	1.0472	1.0472
C	x_{5f}	0.358629	2.85292	1	1.23451	2.20494	1	2
A	x_{6f}	0.895779	2.80957	0.5	0.923094	2.36852	0.5	3



Rysunek 4.14: Sprawdzenie powtarzalności dla algorytmów typu jacobianu rozszerzonego (z lewej) i jacobianu pseudoodwrotnego (z prawej)



Rysunek 4.15: Sprawdzenie powtarzalności dla algorytmów typu jacobianu rozszerzonego (z lewej) i jacobianu pseudoodwrotnego (z prawej)

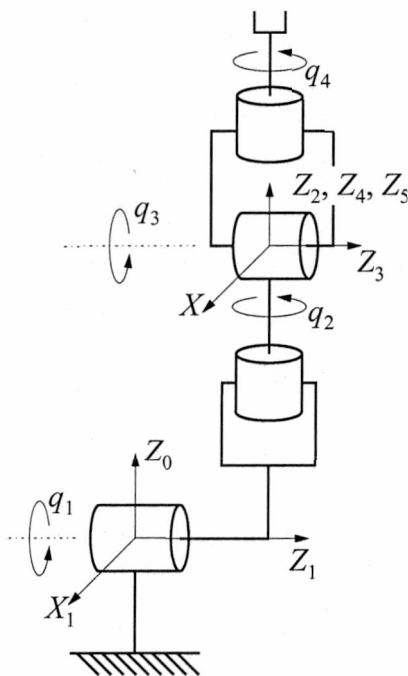


Rysunek 4.16: Sprawdzenie powtarzalności dla algorytmów typu jakobianu rozszerzonego (z lewej) i jakobianu pseudoodwrotnego (z prawej)

4.3 Metoda wariacyjna — Funkcjonał błędu I dla kinematyki w $SE(3)$

4.3.1 Podmanipulator stanfordzki

Dla ilustracji konstrukcji algorytmu jakobianu rozszerzonego dla kinematyki zdefiniowanej jako odwzorowanie w $SE(3)$, rozwiążemy odwrotne zadanie kinematyki dla redundantnego nadgarstka będącego zmodyfikowanym manipulatorem stanfordzkim o czterech stopniach swobody, przedstawionego na rysunku 4.17.



Rysunek 4.17: Redundantny nadgarstek

Orientacja nadgarstka w przestrzeni jest określona za pomocą macierzy rotacji

$$R(x) = \begin{bmatrix} -c_4s_1s_3 + c_1(c_2c_3c_4 - s_2s_4) & s_1s_3s_4 - c_1(c_4s_2 + c_2c_3s_4) & c_3s_1 + c_1c_2s_3 \\ c_3c_4s_2 + c_2s_4 & c_2c_4 - c_3s_2s_4 & s_2s_3 \\ s_1(s_2s_4 - c_2c_3c_4) - c_1c_4s_3 & c_4s_1s_2 + s_4(c_2c_3s_1 + c_1s_3) & c_1c_3 - c_2s_1s_3 \end{bmatrix},$$

gdzie $x = (x_1, \dots, x_4)$ oraz s_i i c_i oznaczają odpowiednio $\sin x_i$ i $\cos x_i$.

Naszym celem jest wyznaczenie funkcji rozszerzającej, która będzie minimalizowała funkcjonal błądu $\mathcal{E}_2$ określony równaniem (2.36). W tym celu również posłużymy się metodą Ritza, omówioną w podrozdziale 2.4.2. Załóżmy dwie postacie funkcji rozszerzającej

- liniową $h_1(x) = a_1x_1 + a_2x_2 + a_3x_3 + a_4x_4$,
- kwadratową $h_2(x) = a_1x_1 + a_2x_2 + a_3x_3 + a_4x_4 + a_5x_1x_2 + a_6x_1x_3 + a_7x_1x_4 + a_8x_2x_3 + a_9x_2x_4 + a_{10}x_3x_4 + a_{11}x_1^2 + a_{12}x_2^2 + a_{13}x_3^2 + a_{14}x_4^2$.

Nieosobliwy obszar w przestrzeni konfiguracyjnej wybierzmy jako

$$\mathbb{Q} = \left\{ x \in \mathbb{R}^4 \mid -\frac{\pi}{2} \leq x_1 \leq \frac{\pi}{2}, 0.01 \leq x_2 \leq \frac{\pi}{2}, 0 \leq x_3 \leq \frac{\pi}{2}, -\frac{\pi}{2} \leq x_4 \leq \frac{\pi}{2} \right\}.$$

Po dokonaniu optymalizacji otrzymaliśmy optymalną liniową funkcję rozszerzającą

$$h_1(x) = -0.454353x_1 - 0.237304x_2 + 0.231516x_3 + 0.464759x_4$$

i kwadratową postać optymalnej funkcji rozszerzającej

$$\begin{aligned} h_2(x) = & -0.46179x_1 + 0.13658x_2 - 0.15069x_3 + 0.46637x_4 + 0.03601x_1x_2 - \\ & 0.09928x_1x_3 - 0.00057x_1x_4 + 0.00524x_2x_3 + 0.09756x_2x_4 - 0.03657x_3x_4 - \\ & 0.01898x_1^2 - 0.24783x_2^2 + 0.25060x_3^2 + 0.01804x_4^2. \end{aligned}$$

Wartość funkcjonalu błędu dla przypadku liniowego jest równa $\mathcal{E}_2(h_1) = 10.0521$, natomiast dla funkcji kwadratowej $\mathcal{E}_2(h_2) = 5.38958$ [22]. Posługując się powyższymi funkcjami rozszerzającymi zbudowano algorytmy typu jakobianu rozszerzonego oznaczone odpowiednio $J_1^{E\#}(x)$ dla funkcji liniowej i $J_2^{E\#}(x)$ dla funkcji kwadratowej.

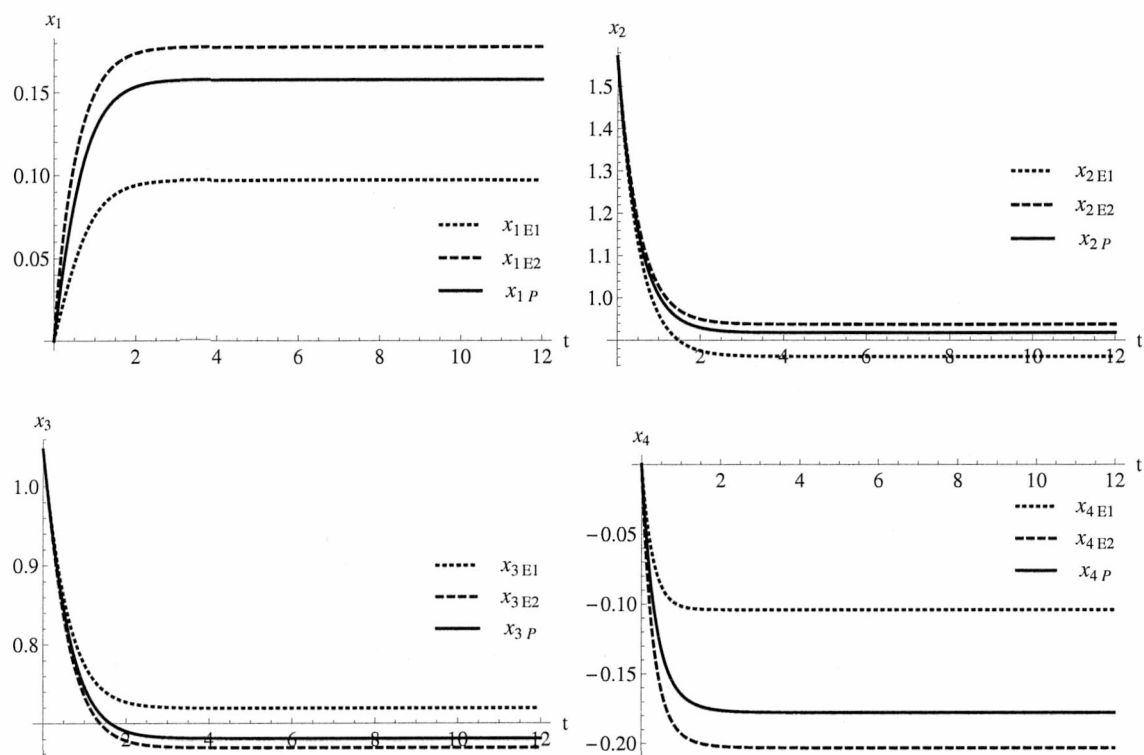
Zachowanie algorytmów kinematyki odwrotnej korzystających z otrzymanych jakobianów rozszerzonych, w porównaniu do algorytmu jakobianu pseudoodwrotnego została przedstawiona na rysunkach 4.18–4.20. W tabeli 4.6 zebrano wartości miar jakości aproksymacji (4.11) i (4.12). Rysunek 4.20 przedstawia kwadrat normy prędkości chwilowych w czasie dla wszystkich algorytmów. W obliczeniach przyjęto odpowiednio:

- Zadanie 1: konfiguracja początkowa $x(0) = (0, \frac{\pi}{2}, \frac{\pi}{3}, 0)$, orientacja zadana

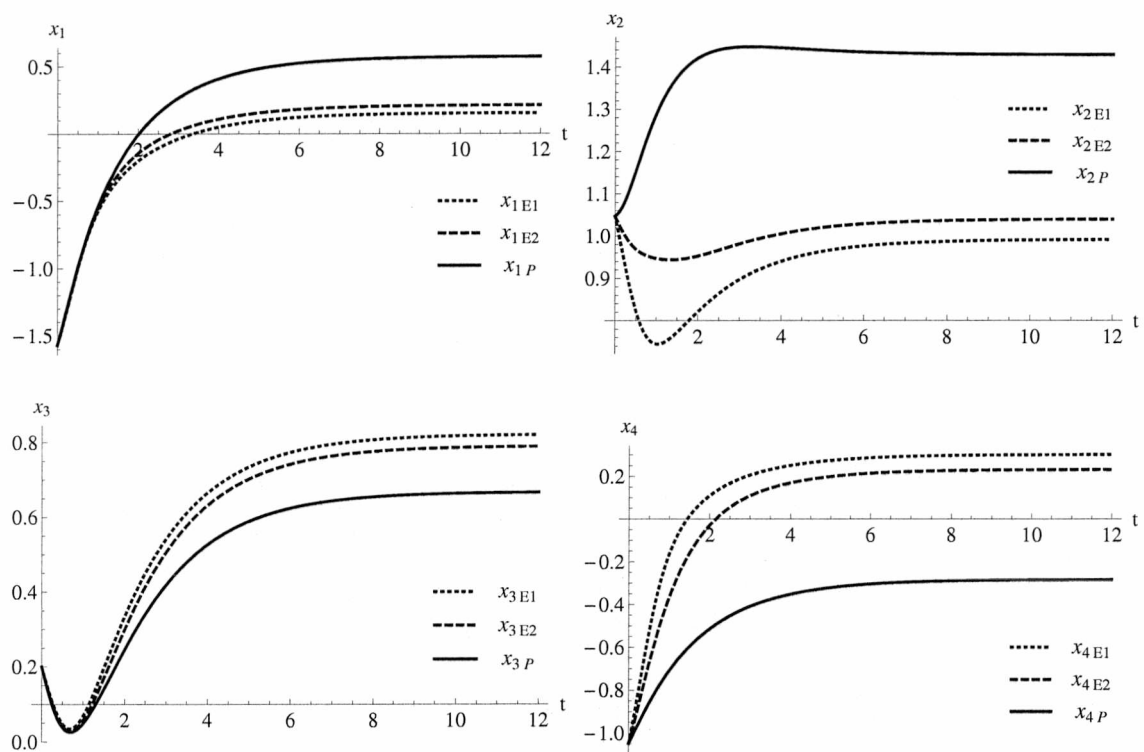
$$R_d = \begin{bmatrix} \frac{1}{2} & -\frac{\sqrt{2}}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{\sqrt{2}}{2} & \frac{1}{2} \\ -\frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} \end{bmatrix},$$

która odpowiada kątom $RPY(\frac{\pi}{4}, \frac{\pi}{4}, 0)$ w reprezentacji Roll–Pitch–Yaw. Współczynnik zbieżności algorytmu $\gamma = 2$ i czas symulacji $t_{\max} = 12$.

- Zadanie 2: konfiguracja początkowa $x(0) = (-\frac{\pi}{2}, \frac{\pi}{3}, 0.2, -\frac{\pi}{3})$, orientacja zadana w kątach Roll–Pitch–Yaw $RPY(\frac{\pi}{2}, \frac{\pi}{4}, \frac{\pi}{6})$. Współczynnik zbieżności algorytmu $\gamma = 0.6$ i czas symulacji $t_{\max} = 12$.



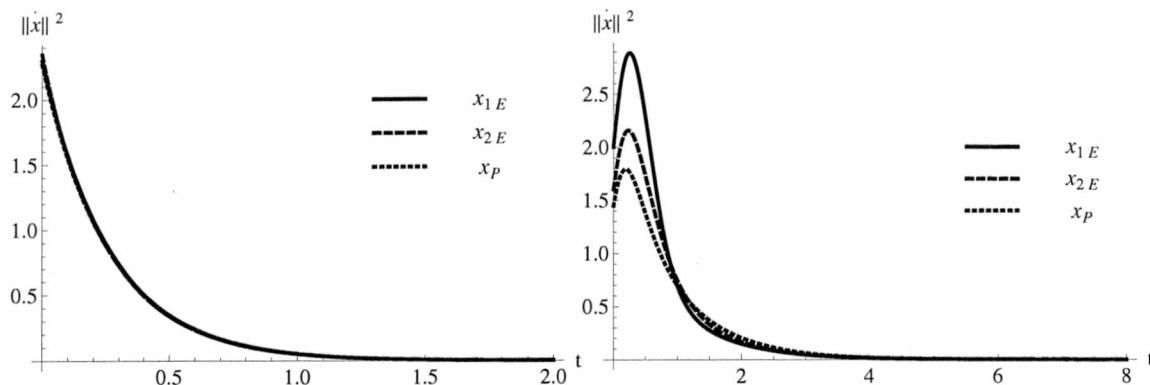
Rysunek 4.18: Rozwiązanie zadania 1



Rysunek 4.19: Rozwiązanie zadania 2

Tabela. 4.6: Wartości miar jakości aproksymacji dla zadań 1 i 2

	Jakobian rozszerzony $J_1^{E\#}$		Jakobian rozszerzony $J_2^{E\#}$	
	δ_1	δ_2	δ_1	δ_2
Zadanie 1	0.09415	0.14098	0.03319	0.10536
Zadanie 2	0.50389	0.61421	0.42605	0.41112



Rysunek 4.20: Normy prędkości przegubowych dla zadania 1 (z lewej) i 2 (z prawej)

Jakość aproksymacji dla algorytmu typu jacobianu rozszerzonego $J_2^{E\#}(x)$ z kwadratową funkcją rozszerzającą jest lepsza niż dla przypadku algorytmu typu jacobianu rozszerzonego $J_1^{E\#}(x)$ z liniową funkcją rozszerzającą (por. rysunki 4.18–4.20 oraz tabela 4.6). Jeśli jednak zależy nam tylko na minimalizacji prędkości ruchu, w większości przypadków oba algorytmy $J_1^{E\#}(x)$ i $J_2^{E\#}(x)$ dają zadowalającą aproksymację algorytmu jacobianu pseudoodwrotnego $J^{P\#}(x)$, co widać na rysunku 4.20.

W celu sprawdzenia powtarzalności nowo otrzymanych algorytmów zażądaliśmy, aby koniec efektora przemieszczał się pomiędzy trzema konfiguracjami w przestrzeni zadaniowej $A = RPY(\frac{\pi}{2}, \frac{\pi}{3}, 0)$, $B = RPY(\frac{\pi}{4}, \frac{\pi}{4}, 0)$, $C = RPY(\frac{\pi}{3}, \frac{\pi}{2}, \frac{\pi}{4})$. Ustalone pozycje przegubowe otrzymane przez wszystkie algorytmy zawarte są w tabeli 4.7.

Tabela. 4.7: Sprawdzenie powtarzalności

Algorytm	Jakobian rozszerzony $J_1^{E\#}(x)$				
A x_0	0	1.5708	1.0472	0	
B x_{1f}	0.0969	0.8614	0.7194	-0.1041	
C x_{2f}	0.5227	0.2999	1.0676	-0.1480	
A x_{3f}	0	1.5708	1.0472	0	
Algorytm	Jakobian rozszerzony $J_2^{E\#}(x)$				
A x_0	0	1.5708	1.0472	0	
B x_{1f}	0.1779	0.9376	0.6690	-0.2031	
C x_{2f}	0.5734	0.3087	1.0191	-0.1656	
A x_{3f}	0	1.5708	1.0472	0	
Algorytm	Jakobian pseudoodwrotny $J^{P\#}(x)$				
A x_0	0	1.5708	1.0472	0	
B x_{1f}	0.1580	0.9177	0.6809	-0.1776	
C x_{2f}	0.6558	0.3260	0.9409	-0.1965	
A x_{3f}	0.1833	1.6756	1.0568	-0.2109	

4.3.2 Manipulator POLYCRANK

Przejdźmy teraz do odwrotnego zadania kinematyki, w którym będziemy rozpatrywać nie tylko orientację, ale także położenie końca efektora. Jako przykład do obliczeń wykorzystamy manipulator POLYCRANK, omówiony w podrozdziale 4.1. Korzystając z parametrów Denavita–Hartenberga, tabela 4.1, otrzymujemy kinematykę manipulatora wyrażoną w $SE(3)$ (4.1). Ponieważ stopień redundancji manipulatora $s = 1$, szukamy jednej funkcji rozszerzającej $h_1(x)$, która będzie minimalizowała błąd aproksymacji $\mathcal{E}_2(h)$ dany równaniem (2.36). Korzystając z metody Ritza, założyliśmy liniową postać funkcji $h(x) = a^\top x$, $a \in \mathbb{R}^7$. Obszar całkowania jest określony przez (4.9). W wyniku obliczeń wykonanych zgodnie z równaniami (2.56) i (2.57), optymalna funkcja rozszerzająca przyjmuje postać

$$h_1(x) = -0.013179x_1 - 0.014623x_2 + 0.021448x_3 - 0.033563x_4 + 0.033571x_5, \quad (4.13)$$

przy wartości funkcjonału $\mathcal{E}_2(h_1) = 0.489715$. Funkcja (4.13) określa jacobian rozszerzony $J_1^{E\#}(x)$ i algorytm jacobianu rozszerzonego (2.35).

Wyniki obliczeń numerycznych dla algorytmów typu jacobianu rozszerzonego i jacobianu pseudoodwrotnego zostały pokazane na rysunku 4.21. Na rysunku 4.22 przedstawiono porównanie normy prędkości przegubowych w trakcie trwania ruchu dla obu algorytmów. Do obliczeń numerycznych przyjęto następujące dane: konfiguracja początkowa $x(0) =$

$$(0.15, 3, 1, 1.5, 3, 0.3, 3), \text{ zadane orientacja końca efektora } R_d = \begin{bmatrix} -1/2 & 0 & \sqrt{3}/2 \\ 0 & -1 & 0 \\ \sqrt{3}/2 & 0 & 1/2 \end{bmatrix}$$

i położenie $T_d = (0.25, 0.65, 1.7)$, współczynnik zbieżności $\gamma = 3$ i czas symulacji $t_{\max} = 5$.

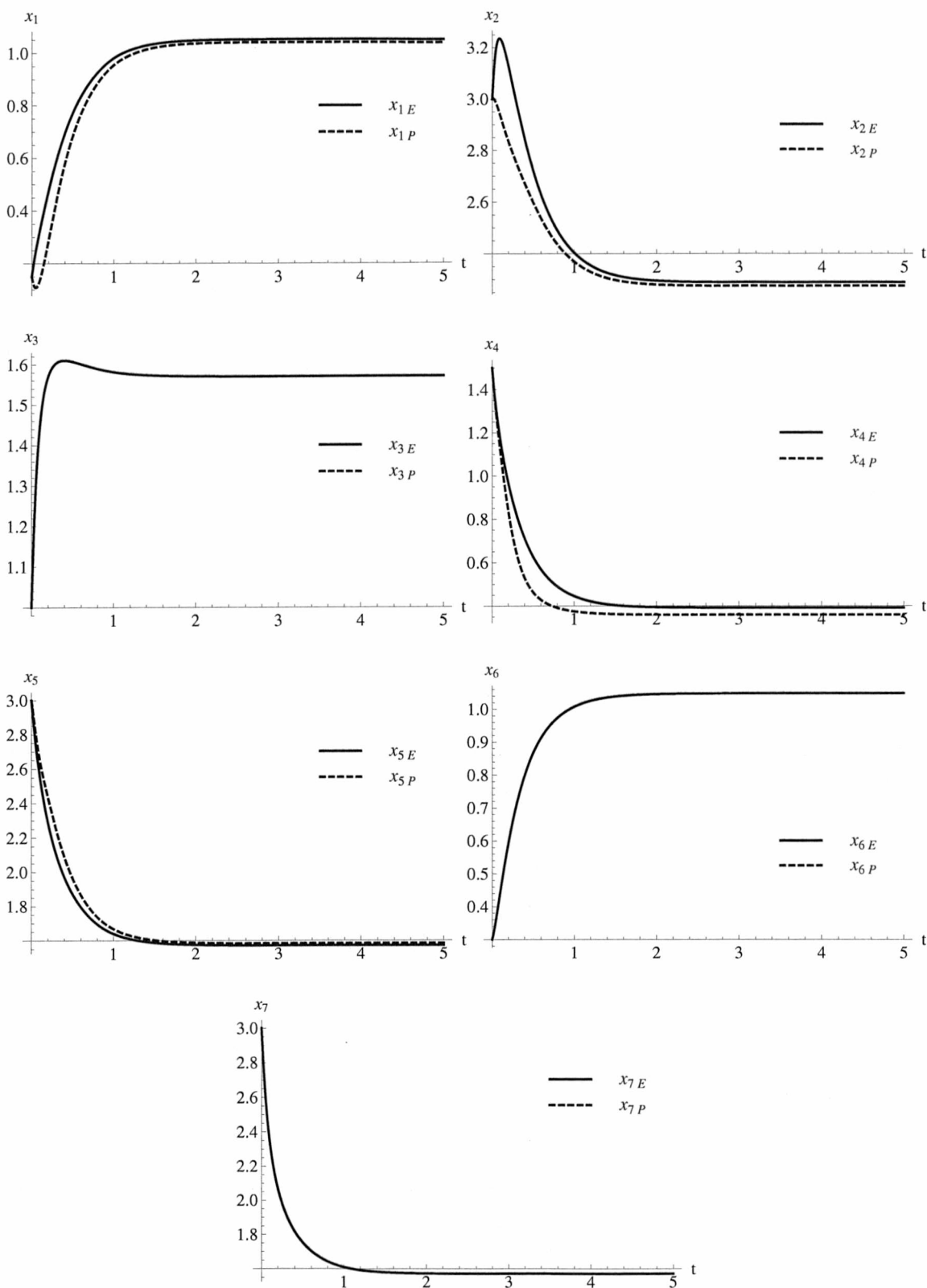
Teraz, zawężmy obszar całkowania do podzbioru konfiguracji wynikających z rozwiązania algorytmu typu jacobianu pseudoodwrotnego w zadaniu 1. W efekcie otrzymamy zadanie optymalizacji na mniejszym obszarze, a minimalna funkcja rozszerzająca i odpowiednio jacobian rozszerzony będą dedykowane ruchowi z zadania 1. Spodziewamy się, że tak otrzymany jacobian rozszerzony zwróci rozwiązanie bliższe rozwiązaniu uzyskanego przy pomocy algorytmu jacobianu pseudoodwrotnego, jednak w taki sposób tracimy na ogólności (zadanie aproksymacji algorytmów staje się zależne od ruchu, który chcemy uzyskać). Założmy, że nowy obszar całkowania jest równy

$$\bar{\mathbb{Q}} = \left\{ x \in \mathbb{R}^7 \mid 0.1 \leq x_1 \leq 1.1, 2.1 \leq x_2 \leq 3.1, 0.9 \leq x_3 \leq 1.7, 0.2 \leq x_4 \leq 1.4, \right. \\ \left. 1.4 \leq x_5 \leq 3.1, 0.2 \leq x_6 \leq 1.2, 1.4 \leq x_7 \leq 3 \right\}.$$

Dla tak obranego obszaru optymalna liniowa funkcja rozszerzająca minimalizująca funkcjonal błędu (2.36) przybiera postać

$$h_2(x) = -0.0747196x_1 - 0.0634196x_2 - 0.00144952x_3 - 0.0812601x_4 + 0.0895736x_5.$$

Rysunek 4.24 przedstawia porównanie rozwiązań (dla danych z zadania 1) dostarczonych przez algorytmy typu jacobianu pseudoodwrotnego i jacobianu rozszerzonego z funkcją rozszerzającą $h_2(x)$. Dodatkowo, na rysunku 4.23 wykreślono kwadraty norm prędkości przegubowych wygenerowanych przez oba algorytmy. Jak widać, średnie prędkości przegubowe otrzymane z algorytmu typu jacobianu rozszerzonego są teraz bliższe średnim prędkościom wygenerowanym przez algorytm typu jacobianu pseudoodwrotnego niż te otrzymane dla algorytmu korzystającego z (4.13). Na potwierdzenie powyższej obserwacji, w tabeli 4.8 zebrano wartości miar jakości aproksymacji (4.11) i (4.12), zaś na rysunkach 4.25 i 4.26 widnieją przebiegi prędkości przegubowych w czasie dla obu algorytmów

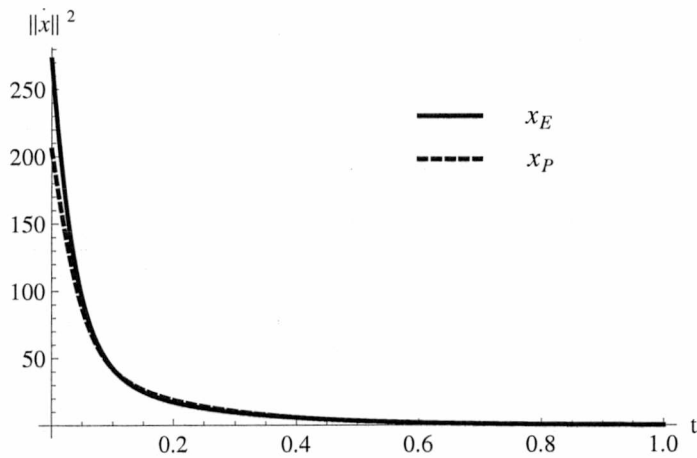


Rysunek 4.21: Rozwiązanie zadania 1 dla algorytmów typu jakobianu pseudoodwrotnego i jakobianu rozszerzonego $J_1^{E\#}$

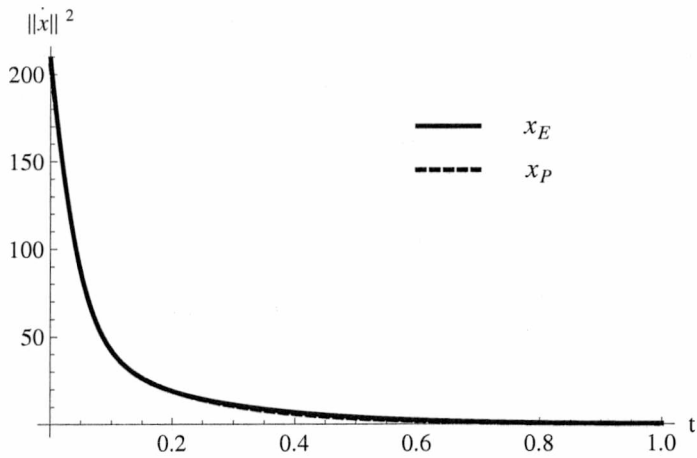
jakobianu rozszerzonego. Oczywiście, aproksymacja nie gwarantuje, że otrzymany optymalny algorytm typu jakobianu rozszerzonego będzie zwracał rozwiązania mieszczące się w założonych przedziałach zmiennych przegubowych. Wynika to z faktu iż rozwiązania generowane przez oba algorytmy w ogólności różnią się ze względu na posiadanie bądź nieposiadanie własności powtarzalności.

Tabela. 4.8: Wartości miar jakości aproksymacji

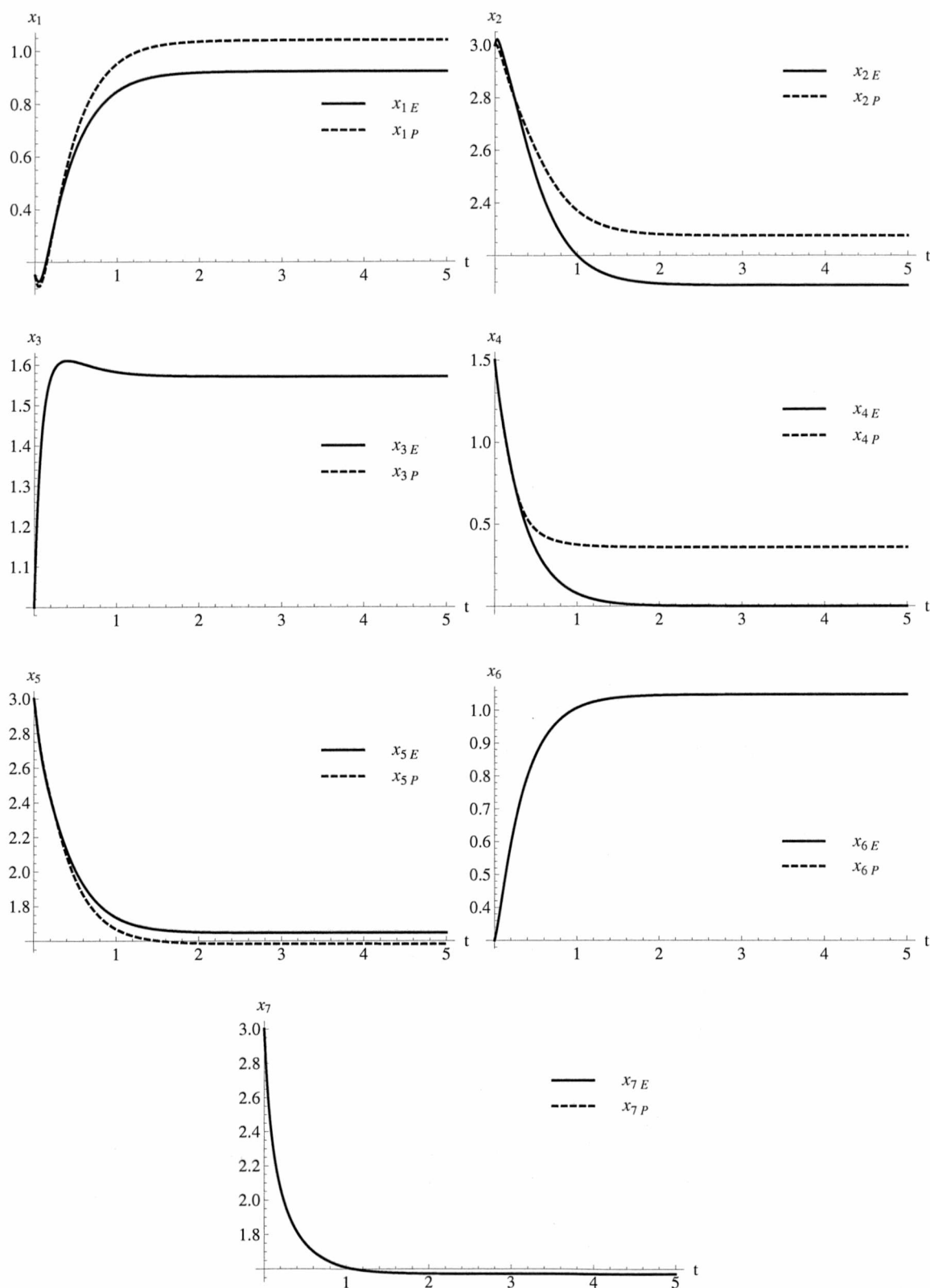
	Jakobian rozszerzony $J_1^{E\#}$		Jakobian rozszerzony $J_2^{E\#}$	
	δ_1	δ_2	δ_1	δ_2
Zadanie 1	0.03092	0.36787	0.09801	0.12950



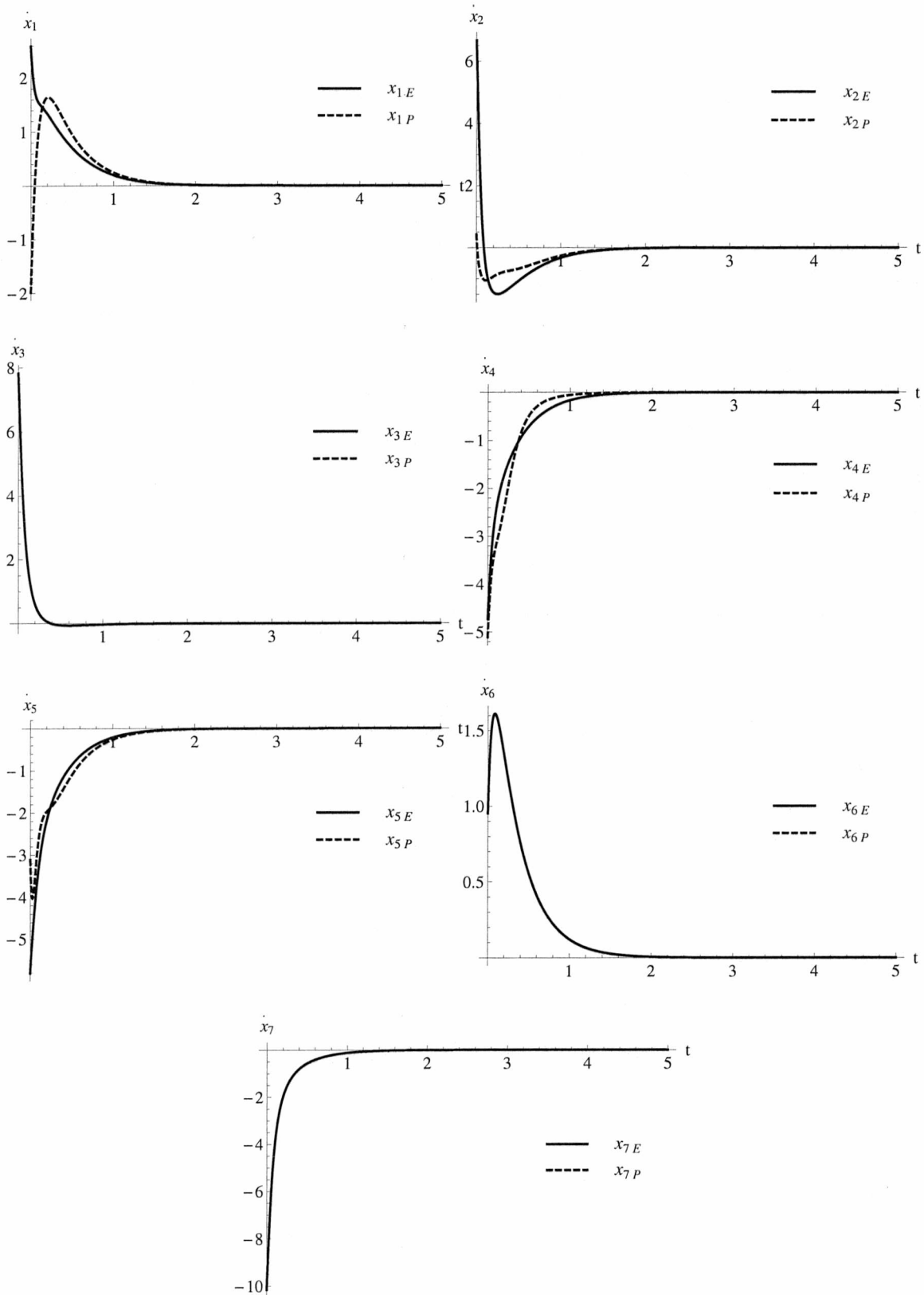
Rysunek 4.22: Normy prędkości przegubowych dla odwrotności $J^{P\#}$ i $J_1^{E\#}$

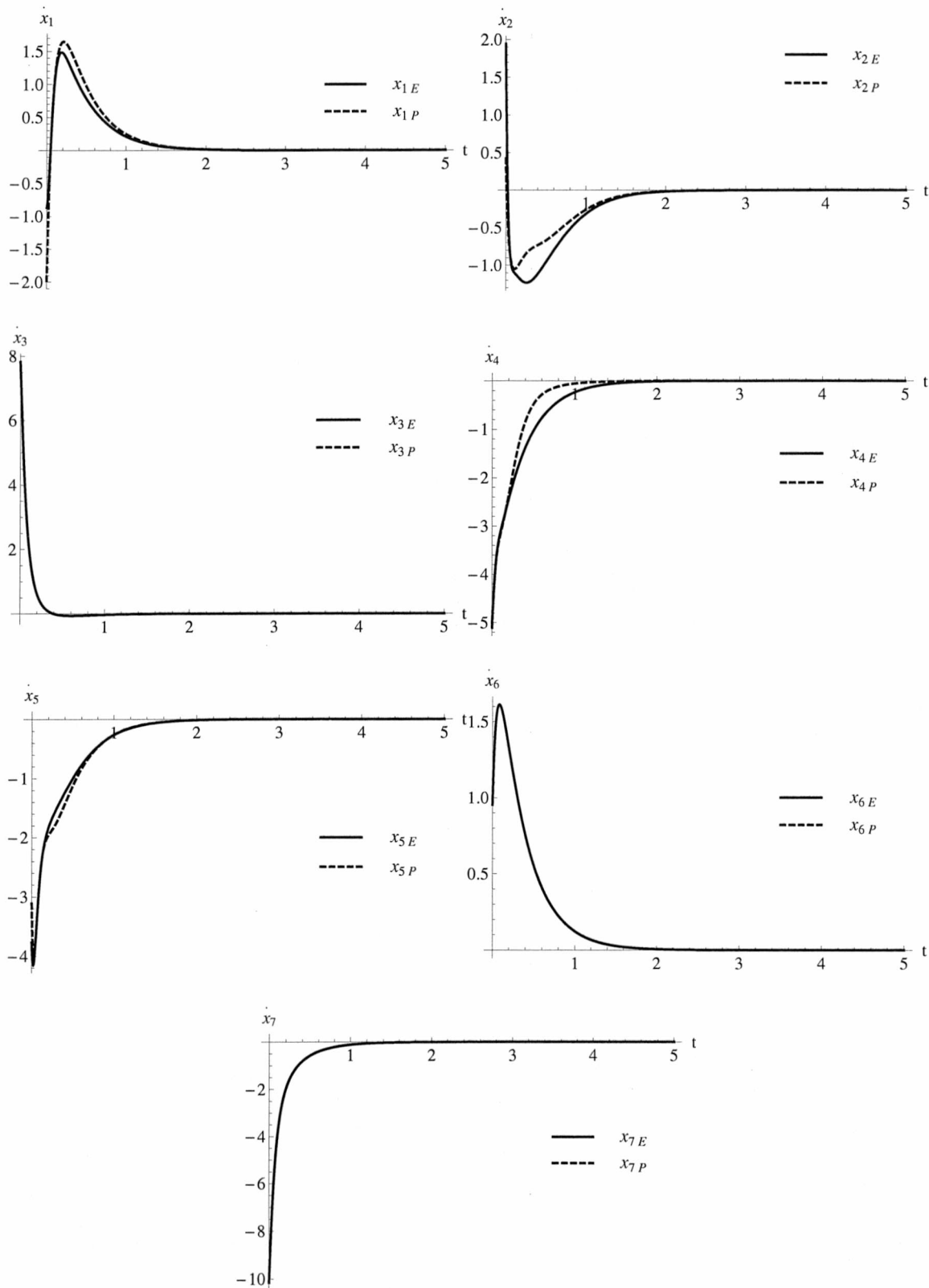


Rysunek 4.23: Normy prędkości przegubowych dla odwrotności $J^{P\#}$ i $J_2^{E\#}$



Rysunek 4.24: Rozwiązanie zadania 2 dla algorytmu typu jakobianu pseudoodwrotnego i jakobianu rozszerzonego $J_2^{E\#}$

Rysunek 4.25: Przebiegi prędkości przegubowych w czasie dla $J^{P\#}$ i $J_1^{E\#}$ (zadanie 1)



Rysunek 4.26: Przebiegi prędkości przegubowych w czasie dla $J^{P\#}$ i $J_2^{E\#}$ (zadanie 2)

4.4 Metoda wariacyjna — Funkcjonał błędu II

4.4.1 Manipulator POLYCRANK

Przyjrzyjmy się teraz zadaniu aproksymacji określonego przez funkcyjonał błędu (2.37) i zastosujmy go dla manipulatora o 7 stopniach swobody POLYCRANK. Kinematyka robota jest zdefiniowana równaniem (4.2). Stopień redundancji robota $s = 1$, zatem szukamy funkcji rozszerzającej $h(x)$, minimalizującej funkcyjonał błędu $\mathcal{E}_3(h)$. Niech funkcja rozszerzająca będzie funkcją liniową $h(x) = a^\top x$, $a \in \mathbb{R}^7$, natomiast obszar całkowania będzie równy

$$\mathbb{Q} = \left\{ x \in \mathbb{R}^7 \mid 0.001 \leq x_1 \leq \frac{\pi}{2}, \frac{\pi}{2} + 0.001 \leq x_2 \leq \pi, 0 \leq x_3 \leq \pi, 0.001 \leq x_4 \leq \frac{\pi}{2}, \right. \\ \left. \frac{\pi}{2} + 0.001 \leq x_5 \leq \pi - 0.001, 0.001 \leq x_6 \leq \frac{\pi}{2}, 0 \leq x_7 \leq \pi \right\}.$$

Dla tak założonych danych optymalna funkcja rozszerzająca przybiera postać

$$h(x) = -0.195322x_1 - 0.2918x_2 - 0.504088x_4 + 0.504256x_5.$$

Za pomocą tak zdefiniowanej funkcji rozszerzającej możemy zdefiniować jacobian rozszerzony oraz zdefiniować algorytm typu jacobianu rozszerzonego. Obliczenia przeprowadzono dla dwóch zadań. Zadanie pierwsze jest opisane warunkiem początkowym $x_0 = (0.3, 1.6, 1.2, 1, 2, 0.5, 0.5)$ i warunkiem końcowym $y_d = (0.21, 0.51, 1.5, \frac{\pi}{2}, \frac{\pi}{3}, \frac{\pi}{3})$. Natomiast, drugie zadanie: $x_0 = (1, 1.7, 0.2, 1.5, 3, 0.5, 3)$ i $y_d = (0.21, 0.61, 1.7, \frac{\pi}{2}, \frac{\pi}{3}, \frac{\pi}{3})$. Pozostałe parametry potrzebne do obliczeń były takie same dla obu zadań i były równe: współczynnik zbieżności $\gamma = 3$, czas symulacji $t_{\max} = 5$.

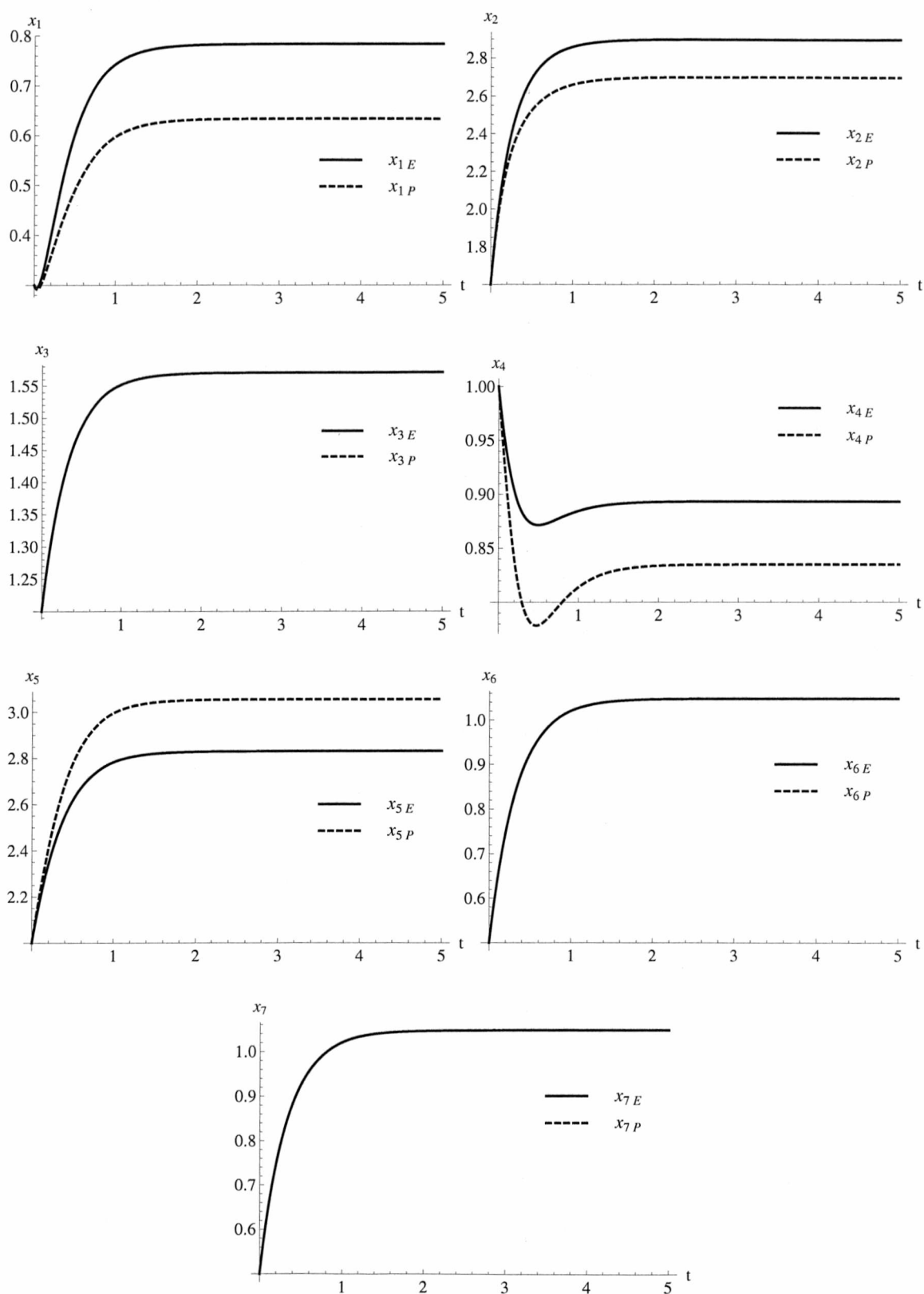
Rozwiązania powyższych zadań zostały przedstawione na rysunkach 4.27 i 4.28. Rysunki 4.29 i 4.30 przedstawiają przebiegi normy prędkości przegubowych w czasie. Natomiast tabela 4.9 zawiera wartości miar jakości aproksymacji (4.11) i (4.12).

Tabela. 4.9: Wartości miar jakości aproksymacji

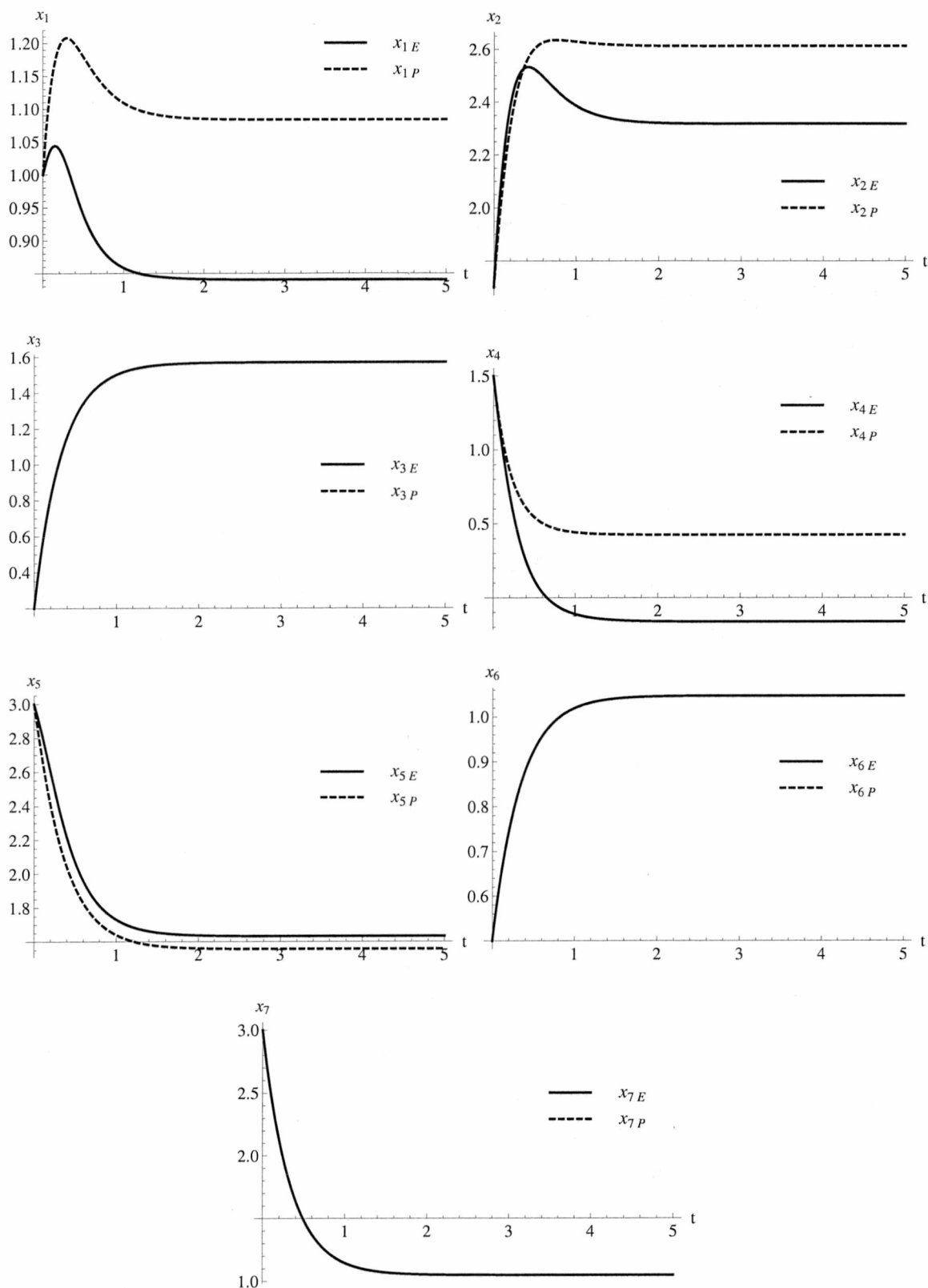
	Jacobian rozszerzony $J_1^{E\#}$	
	δ_1	δ_2
Zadanie 1	0.07037	0.18430
Zadanie 2	0.16762	0.25195

Analizując wyniki widać, że zadanie aproksymacji zostało rozwiązane poprawnie. Rozwiązania otrzymane dzięki algorytmowi jacobianu rozszerzonego są bliskie rozwiązaniom algorytmu jacobianu pseudoodwrotnego. Także wartości norm prędkości przegubowych rozwiązań obu algorytmów są do siebie zbliżone.

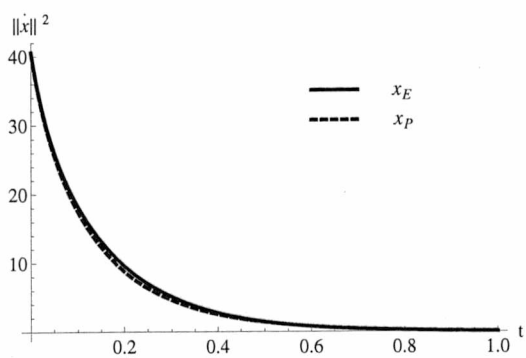
W celu sprawdzenia powtarzalności otrzymanego algorytmu typu jacobianu rozszerzonego, wyznaczono trzy punkty w przestrzeni zadaniowej $A = (0.21, 0.51, 1.7, \frac{\pi}{2}, \frac{\pi}{3}, \frac{\pi}{3})$, $B = (0.48, 0.22, 1.51, 1, 1, 2)$ i $C = (0.36, 0.20, 1.53, 0.5, 0.5, 3)$. Zadanie polega na zakreśleniu przez koniec efektora krzywej zamkniętej zawartej pomiędzy tymi punktami. Rozwiązanie zadania przez algorytm jacobianu pseudoodwrotnego i jacobianu rozszerzonego zawierają rysunki 4.31–4.33, z których można wywnioskować, że algorytm typu jacobianu rozszerzonego zachowuje cechę powtarzalności, w przeciwieństwie do algorytmu typu jacobianu pseudoodwrotnego.



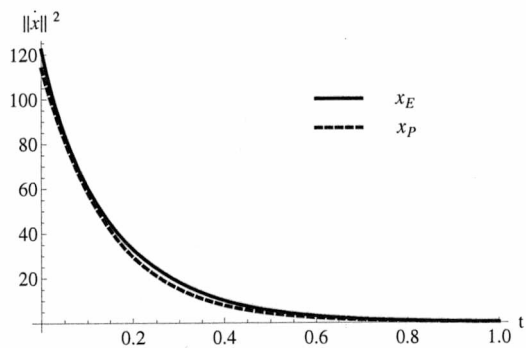
Rysunek 4.27: Rozwiązanie zadania 1 dla algorytmów typu jakobianu pseudoodwrotnego i jakobianu rozszerzonego



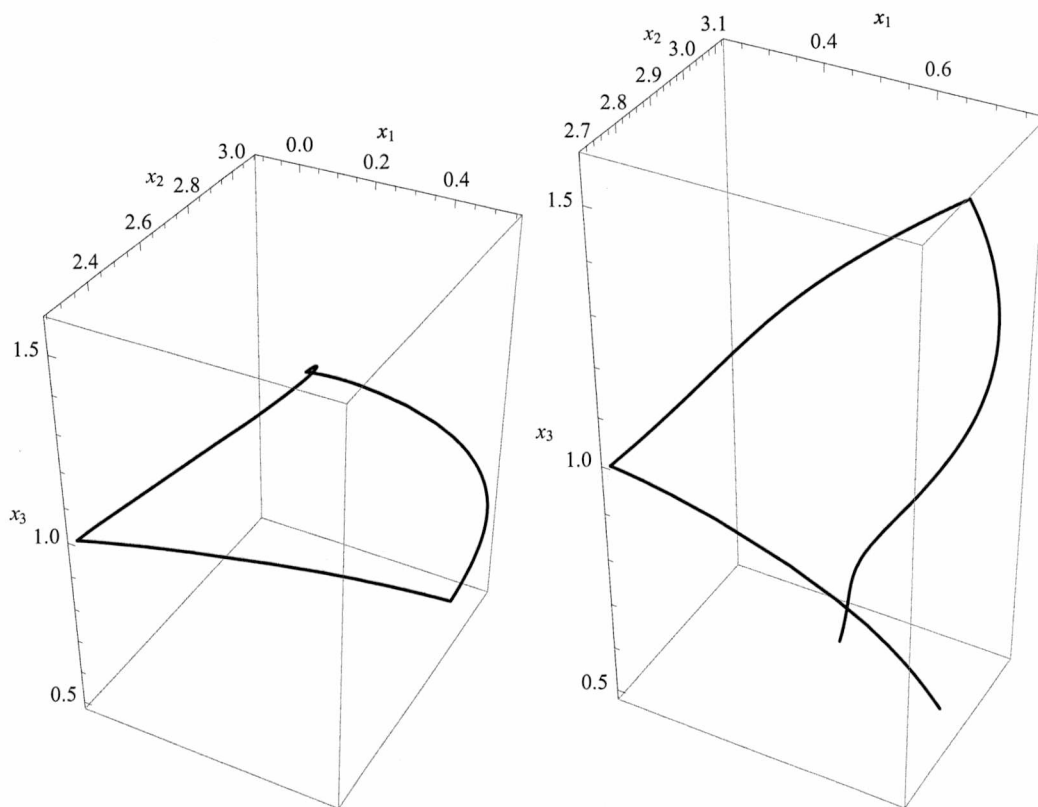
Rysunek 4.28: Rozwiązanie zadania 2 dla algorytmów typu jacobianu pseudoodwrotnego i jacobianu rozszerzonego



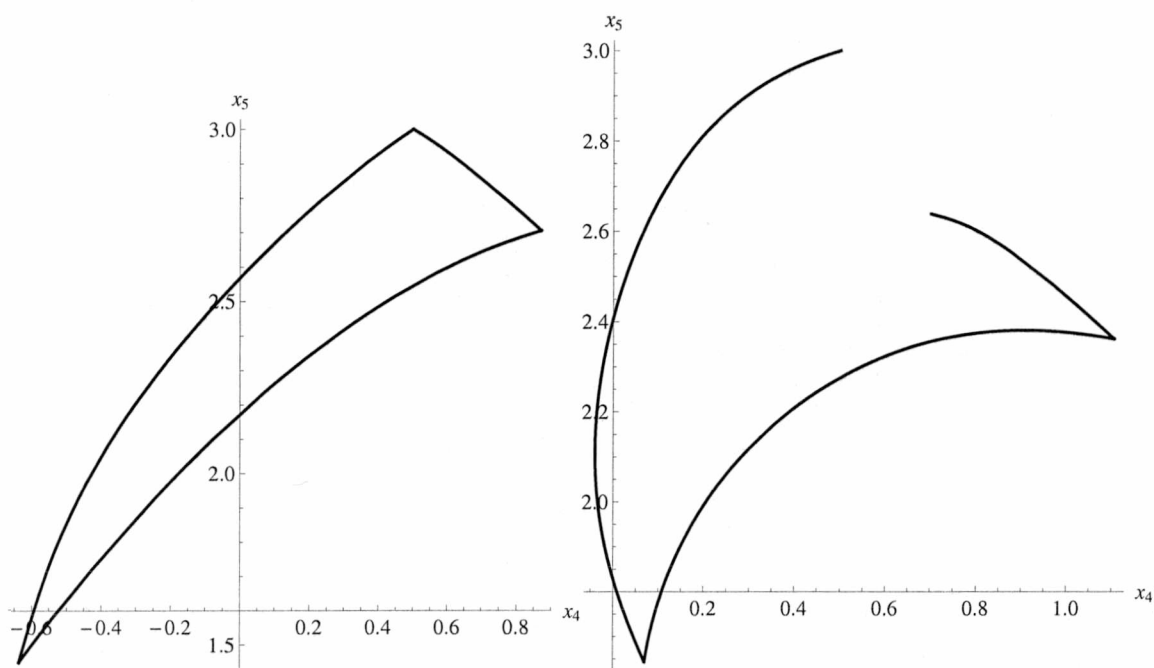
Rysunek 4.29: Normy prędkości przegubowych w zadaniu 1 dla jacobianów $J^{P\#}$ i $J^{E\#}$



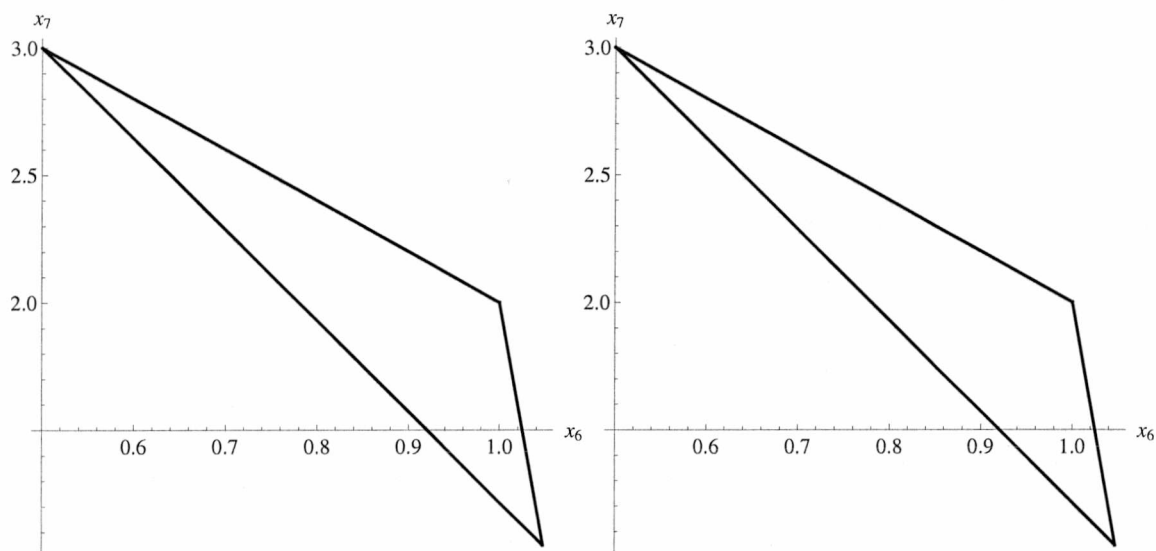
Rysunek 4.30: Normy prędkości przegubowych w zadaniu 2 dla jacobianów $J^{P\#}$ i $J^{E\#}$



Rysunek 4.31: Sprawdzenie powtarzalności dla algorytmów typu jacobianu rozszerzonego (z lewej) i jacobianu pseudoodwrotnego (z prawej)



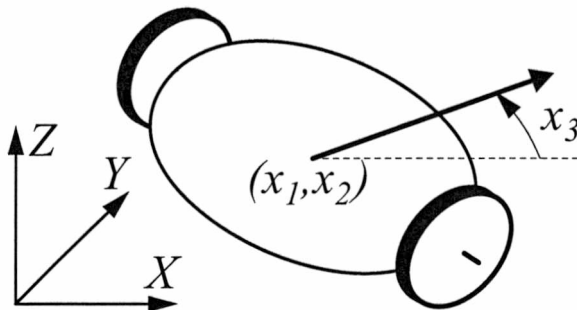
Rysunek 4.32: Sprawdzenie powtarzalności dla algorytmów typu jacobianu rozszerzonego (z lewej) i jacobianu pseudoodwrotnego (z prawej)



Rysunek 4.33: Sprawdzenie powtarzalności dla algorytmów typu jacobianu rozszerzonego (z lewej) i jacobianu pseudoodwrotnego (z prawej)

4.4.2 Monocykl

Rozwiążmy zadanie z błędem aproksymacji (2.37) dla robota mobilnego. Obliczenia numeryczne przeprowadzimy dla kinematyki monocykla. Schemat robota z odpowiednimi oznaczeniami przedstawia rysunek 4.34.



Rysunek 4.34: Monocykl

Niech $x = (x_1, x_2, x_3)$ określa pozycję i orientację monocykla poruszającego się po płaszczyźnie. Kinematyka monocykla jest zdefiniowana za pomocą układu sterowania

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{bmatrix} \cos x_3 & 0 \\ \sin x_3 & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix},$$

gdzie sterowania u_1 i u_2 mają znaczenie prędkości liniowej i kątowej robota. Niech horyzont czasowy symulacji będzie $t_{\max} = T = 2\pi$, a sterowania przyjmą postać

$$\begin{aligned} u_1(t) &= \lambda_{10} + \lambda_{11} \sin t, \\ u_2(t) &= \lambda_{20} + \lambda_{21} \sin t, \end{aligned} \tag{4.14}$$

tak, że wektor współczynników funkcji sterujących $\lambda = (\lambda_{10}, \lambda_{11}, \lambda_{20}, \lambda_{21}) \in \mathbb{R}^4$.

Załóżmy, że warunki początkowe $x_0 = 0$. Wtedy, kinematyka monocykla podlegającego sterowaniu (4.14) jest następująca [56]

$$K_{x_0, T}(\lambda) = (x_{1\lambda}(T), x_{2\lambda}(T), x_{3\lambda}(T)), \tag{4.15}$$

gdzie

$$\begin{aligned} x_{1\lambda}(t) &= \int_0^t (\lambda_{10} + \lambda_{11} \sin s) \cos x_{3\lambda} \, ds, & x_{2\lambda}(t) &= \int_0^t (\lambda_{10} + \lambda_{11} \sin s) \sin x_{3\lambda} \, ds, \\ x_{3\lambda}(t) &= \int_0^t (\lambda_{20} + \lambda_{21} \sin s) \, ds = \lambda_{20}t + \lambda_{21}(1 - \cos t). \end{aligned}$$

Mając daną kinematykę (4.15), obliczamy jacobian

$$J_{x_0, T}(\lambda) = \frac{\partial K_{x_0, T}(\lambda)}{\partial \lambda} = [J_{ij}(\lambda)], \tag{4.16}$$

gdzie

$$\begin{aligned}
J_{11}(\lambda) &= \int_0^T \cos x_{3\lambda}(s) \, ds, & J_{12}(\lambda) &= \int_0^T \sin s \cos x_{3\lambda}(s) \, ds, \\
J_{13}(\lambda) &= - \int_0^T (\lambda_{10} + \lambda_{11} \sin s) \sin x_{3\lambda}(s) s \, ds, \\
J_{14}(\lambda) &= - \int_0^T (\lambda_{10} + \lambda_{11} \sin s) \sin x_{3\lambda}(s) (1 - \cos t) \, ds, \\
J_{21}(\lambda) &= \int_0^T \sin x_{3\lambda}(s) \, ds, & J_{22}(\lambda) &= \int_0^T \sin s \sin x_{3\lambda}(s) \, ds, \\
J_{23}(\lambda) &= - \int_0^T (\lambda_{10} + \lambda_{11} \sin s) \cos x_{3\lambda}(s) s \, ds, \\
J_{24}(\lambda) &= - \int_0^T (\lambda_{10} + \lambda_{11} \sin s) \cos x_{3\lambda}(s) (1 - \cos t) \, ds, \\
J_{31}(\lambda) &= 0, & J_{32}(\lambda) &= 0, & J_{33}(\lambda) &= 2\pi, & J_{34}(\lambda) &= 0.
\end{aligned}$$

Taka postać Jakobianu (4.16) pozwala na wyznaczenie jego pseudoodwrotności jako

$$J_{x_0, T}^{P\#}(\lambda) = J_{x_0, T}^T(\lambda) (J_{x_0, T}(\lambda) J_{x_0, T}^T(\lambda))^{-1}. \quad (4.17)$$

Ponieważ $s = 1$, kodystrybucja anihilująca (4.17) jest wyrażona przez 1-formę $\Omega(\lambda) = (\omega_1(\lambda), \omega_2(\lambda), \omega_3(\lambda), \omega_4(\lambda))$, której składnikami są

$$\begin{aligned}
\omega_1(\lambda) &= \frac{J_{14}(\lambda)J_{22}(\lambda) - J_{12}(\lambda)J_{24}(\lambda)}{J_{11}(\lambda)J_{24}(\lambda) - J_{14}(\lambda)J_{21}(\lambda)}, & \omega_2(\lambda) &= 1, \\
\omega_3(\lambda) &= 0, & \omega_4(\lambda) &= \frac{J_{12}(\lambda)J_{21}(\lambda) - J_{11}(\lambda)J_{22}(\lambda)}{J_{11}(\lambda)J_{24}(\lambda) - J_{14}(\lambda)J_{21}(\lambda)}.
\end{aligned}$$

Łatwo zauważyć, że $\omega_1(\lambda)$ i $\omega_4(\lambda)$ są dobrze określone, gdy $\lambda_{10} \neq 0$.

Jesteśmy teraz gotowi, aby zdefiniować zadanie aproksymacji: znaleźć funkcję rozszerzającą $h(x)$ minimalizującą funkcjonal błędu określony równaniem (2.37). Niech obszar całkowania będzie równy

$$\mathbb{Q} = \{\lambda \in \mathbb{R}^4 \mid 0.4 \leq \lambda_{10} \leq 2.5, -0.5 \leq \lambda_{11} \leq 2.5, -0.5 \leq \lambda_{20} \leq 2.5, -0.5 \leq \lambda_{21} \leq 2.5\}.$$

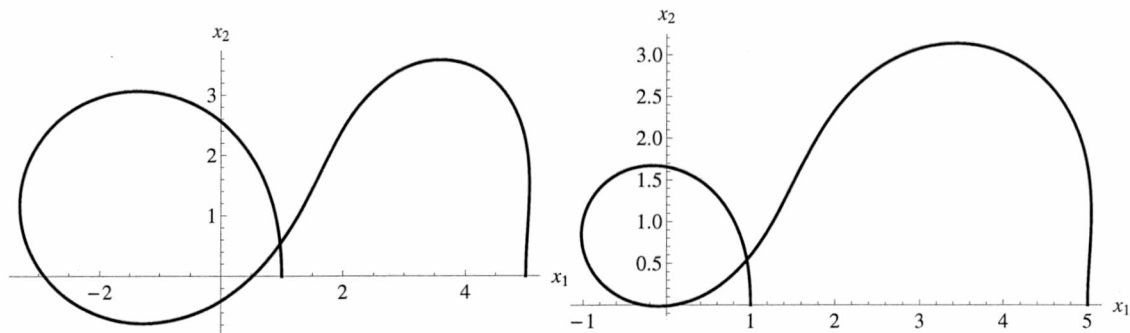
Zakładając liniową postać funkcji rozszerzającej $h(\lambda) = a^T \lambda$, a następnie posługując się metodą Ritza, otrzymaliśmy następujący wynik

$$h(\lambda) = 0.270289\lambda_{10} + \lambda_{11} + 0.159457\lambda_{21}.$$

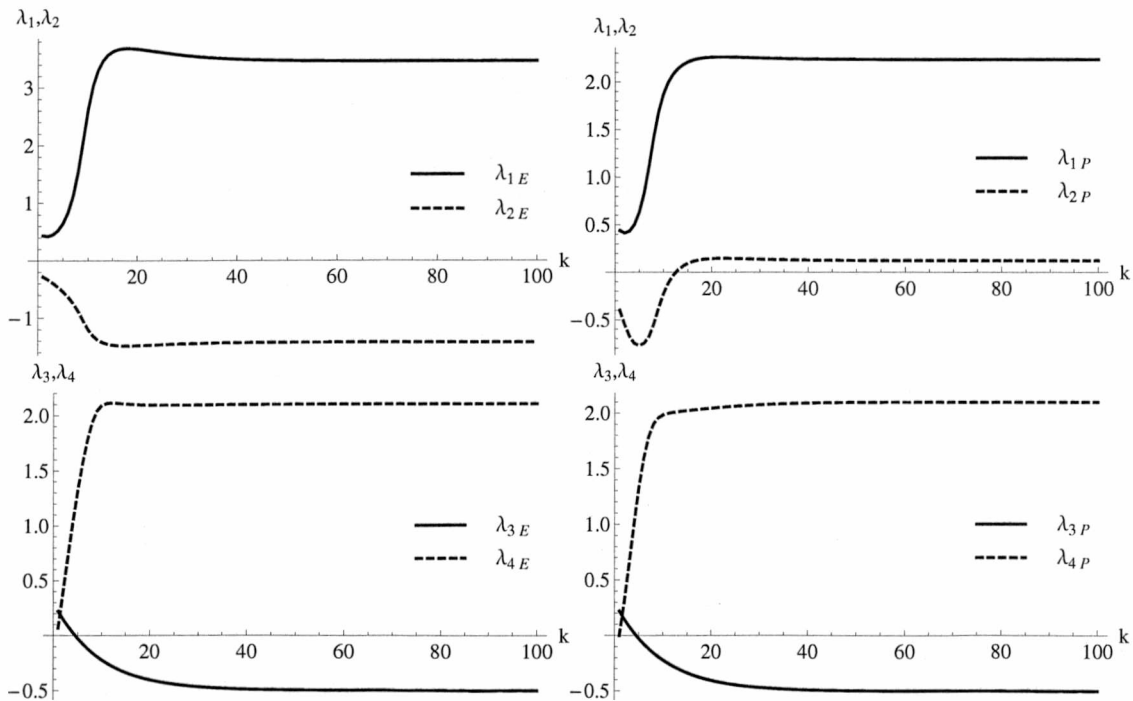
Rozpatrzmy dwa zadania kinematyki odwrotnej o różnych punktach początkowych. Warunek początkowy dla zadania 1 jest równy $x_0 = (5, 0, \frac{\pi}{2})$, natomiast dla 2 $x_0 = (-5, 5, \frac{\pi}{2})$. Pozostałe parametry potrzebne do obliczeń są takie same dla obu przypadków. Są to: czas symulacji $t_{\max} = T = 2\pi$, współczynnik zbieżności algorytmu $\gamma = 0.1$, współczynniki sterowań początkowych $\lambda_0 = (0.5, -0.25, 0.3, -0.25)$ i zadany punkt końcowy w przestrzeni zadaniowej $y_d = (1, 0, -\frac{\pi}{2})$.

Ścieżki (x_1, x_2) w przestrzeni zadaniowej będące rozwiązaniem zadań kinematyki odwrotnej zostały przedstawione na rysunkach: 4.35 dla zadania 1 oraz 4.37 dla zadania 2. Na rysunkach pokazano zarówno rozwiązania dla algorytmu typu jakobianu rozszerzonego, jak i dla algorytmu typu jakobianu pseudoodwrotnego. Rysunki 4.36 i 4.38 pokazują

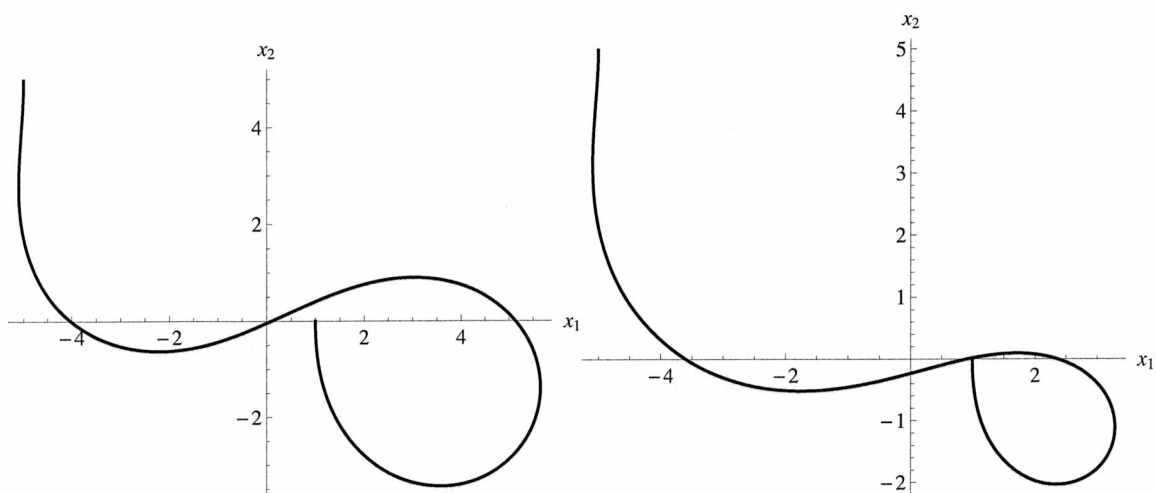
sterowania wygenerowane w kolejnych krokach przez oba algorytmy. Powyższe zadania zostały również rozwiązane w artykule [56], przy zastosowaniu metody geometrycznej. Porównując powyższe wyniki z wynikami zawartymi w artykule, można zauważyć, że ścieżka w przestrzeni zadaniowej zadania 1 uzyskana za pomocą metody geometrycznej jest bliższa rozwiązaniu algorytmu pseudoodwrotnego niż ścieżka otrzymana metodą wariacyjną. Z drugiej strony, porównując wyniki otrzymane dla zadania 2 widać, że bardziej dokładna jest metoda wariacyjna, metoda geometryczna zwróciła ścieżkę znacznie odległą od ścieżki algorytmu typu jacobianu pseudoodwrotnego. Zarówno w pierwszym, jak i w drugim przypadku metoda wariacyjna zwraca ścieżki odległe od ścieżek algorytmu typu jacobianu pseudoodwrotnego, ale o podobnym charakterze.



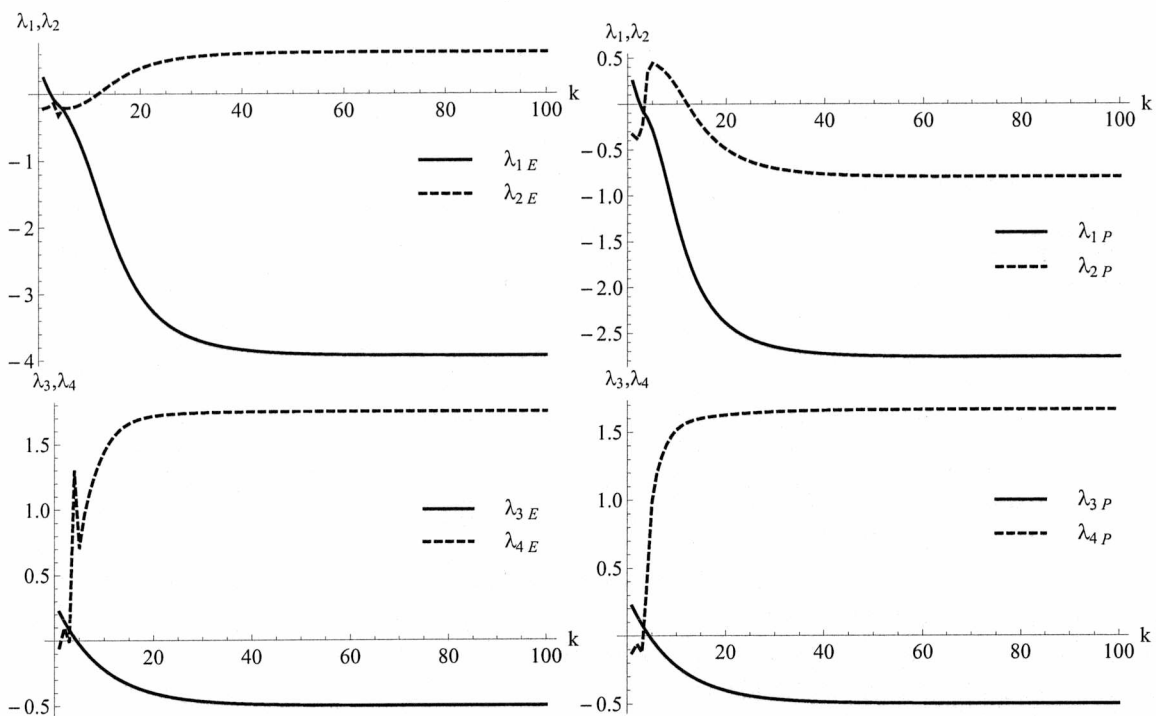
Rysunek 4.35: Ścieżka w przestrzeni zadaniowej dla algorytmów typu jacobianu rozszerzonego (z lewej) i jacobianu pseudoodwrotnego (z prawej) – zadanie 1



Rysunek 4.36: Sterowania λ dla algorytmów typu jacobianu rozszerzonego (z lewej) i jacobianu pseudoodwrotnego (z prawej) – zadanie 1



Rysunek 4.37: Ścieżka w przestrzeni zadaniowej dla algorytmów typu jacobianu rozszerzonego (z lewej) i jacobianu pseudoodwrotnego (z prawej) – zadanie 2



Rysunek 4.38: Sterowania λ dla algorytmów typu jacobianu rozszerzonego (z lewej) i jacobianu pseudoodwrotnego (z prawej) – zadanie 2

4.5 Metoda geometryczna

W niniejszym podrozdziale przedstawimy wyniki obliczeń dla metody geometrycznej opisanej w podrozdziale 2.3. Metoda ta posłuży nam także do porównań z metodą wariacyjną.

4.5.1 Manipulator POLYCRANK

Wyberzmy w przestrzeni przegubowej manipulatora obszar konfiguracji nieosobliwych $\mathbb{Q}$ i zdefiniujmy jego foliację, której liśćmi są proste równoległe do osi współrzędnych x_4 [54]

$$E_{a,b,c,d} = \{\alpha_1 + a\} \times \{\alpha_2 + b\} \times \{\alpha_3\} \times \mathbb{R} \times \{\alpha_5 + c\} \times \{\alpha_6\} \times \{\alpha_7\},$$

parametryzowaną przez $\alpha \in \mathbb{R}^6$ i zależną od parametrów $a, b, c \neq 0$, $a \neq b$, których rolą jest umieszczenie liścia odniesienia $E_{a,b,c,0} = \{a\} \times \{b\} \times \{0\} \times \mathbb{R} \times \{c\} \times \{0\} \times \{0\}$ odpowiednio daleko od konfiguracji osobliwych. Wtedy, homotopia może być zdefiniowana jako

$$\begin{aligned} \Phi_t(x) &= tx + (1-t)(a, b, c, x_4, c, 0)^\top = \\ &= (t(x_1 - a) + a, t(x_2 - b) + b, t(x_5 - c) + c, tx_6, tx_7)^\top. \end{aligned} \quad (4.18)$$

Na tej podstawie obliczamy pole wektorowe (2.38)

$$X(x) = \left. \frac{d\Phi_t(x)}{dt} \right|_{t=1} = \begin{pmatrix} x_1 - a \\ x_2 - b \\ x_3 \\ 0 \\ x_5 - c \\ x_6 \\ x_7 \end{pmatrix}.$$

1-forma anihilująca jacobian transponowany, a jednocześnie zgodna z foliacją jest następująca [54]

$$\Omega = \left(-\frac{l_4 \sin(x_5 - x_4) \sin(x_3 - x_2)}{l_1 \sin x_5 \sin(x_2 - x_1)}, \frac{l_4 \sin(x_5 - x_4) \sin(x_3 - x_1)}{l_2 \sin x_5 \sin(x_2 - x_1)}, 0, 1, -\frac{l_4 \sin x_4}{l_5 \sin x_5}, 0, 0 \right). \quad (4.19)$$

Łącząc tę 1-formę z polem wektorowym (4.18) otrzymujemy

$$F(x) = \Omega(x)X(x) = -\frac{l_2 l_4 l_5 s_{54} s_{32}(x_1 - a) - l_1 l_4 l_5 s_{54} s_{31}(x_2 - b) + l_1 l_2 l_4 s_{421}(x_5 - c)}{l_1 l_2 l_5 s_5 s_{21}},$$

gdzie $s_{ij} = \sin(x_i - x_j)$. Następnie, wprowadźmy nowe współrzędne $x = (y, z)$, takie że $z = x_4$ zmienia się wzdłuż liści oraz $y_i = x_i$, $i = 1, 2, 3$ i $y_{i+3} = x_{i+4}$ dla $i = 1, 2, 3$ są poprzeczne do liści. Podążając za procedurą opisaną w rozdziale 2.3, wyznaczamy równanie charakterystyczne (2.45)

$$\begin{aligned} \frac{dz(t)}{dt} &= \frac{l_4(y_1 - a) \sin(t(y_4 - c) + c - z) \sin(ty_3 - t(y_2 - b) - b)}{l_1 \sin(t(y_4 - c) + c) \sin(t(y_2 - b) + b - t(y_1 - a) - a)} + \\ &= \frac{l_4(y_4 - c) \sin z}{l_5 \sin(t(y_4 - c) + c)} - \frac{l_4(y_2 - b) \sin(t(y_4 - c) + c - z) \sin(ty_3 - t(y_1 - a) - a)}{l_2 \sin(t(y_4 - c) + c) \sin(t(y_2 - b) + b - t(y_1 - a) - a)}. \end{aligned} \quad (4.20)$$

Niech $\varphi(t, y, z_0)$ oznacza strumień (4.20). Funkcja rozszerzająca $h(x) = h(y, z)$ określająca odwrotność jacobianu rozszerzonego spełnia następujące równanie

$$z = \varphi(1, y, h(y, z))$$

i może być obliczona numerycznie. Tak otrzymaną odwrotność jacobianu rozszerzonego zastosujemy do rozwiązywania dwóch zadań kinematyki odwrotnej dla POLYCRANKa. Konfiguracja początkowa dla zadania 1 wynosi $x_0 = (0.8, \frac{3\pi}{4}, \frac{\pi}{2}, 0.1, \frac{2\pi}{3}, \frac{\pi}{6}, \frac{\pi}{2})$, natomiast punkt zadany w przestrzeni zadaniowej $y_d = (0.30, 0.51, 1.60, \frac{\pi}{2}, \frac{\pi}{2}, \frac{\pi}{2})$. Analogicznie, drugie zadanie opisane jest przez $x_0 = (1, \pi, \pi, \frac{\pi}{3}, 2, \frac{\pi}{4}, \frac{\pi}{3})$ oraz $y_d = (-0.38, 0.65, 1.40, \pi, \frac{\pi}{4}, \frac{\pi}{3})$. Liść odniesienia ustalono zakładając $a = \frac{\pi}{4}$ i $b = c = \frac{3\pi}{4}$.

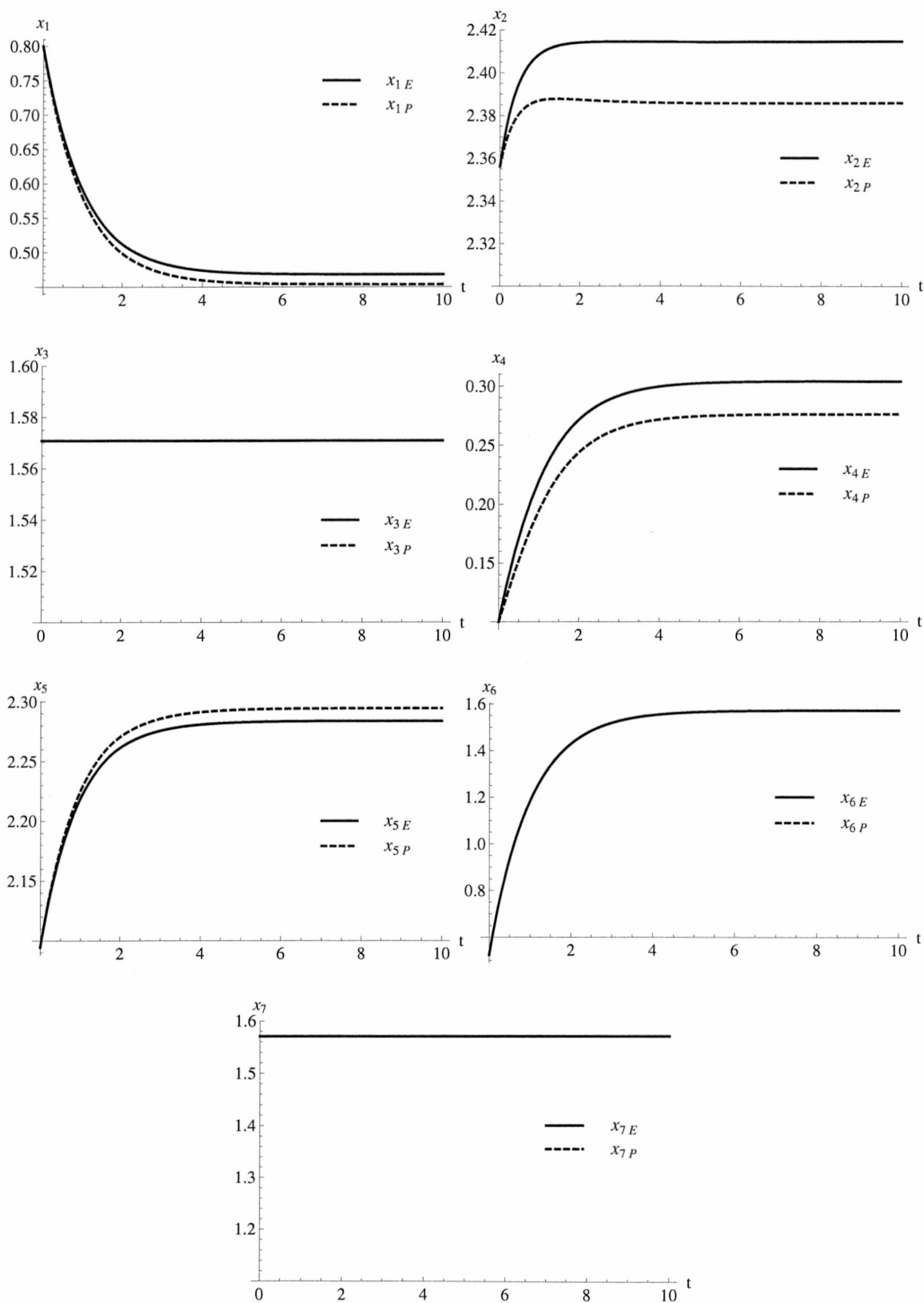
Rozwiązania obu zadań zostały zobrazowane na rysunkach 4.39 i 4.40. Rysunek 4.41 przedstawia kwadrat normy prędkości przegubowych $\|\dot{x}(t)\|^2$ w zależności od czasu dla algorytmów typu jacobianu rozszerzonego i jacobianu pseudoodwrotnego, odpowiednio dla obu zadań.

Z rysunków można wywnioskować, iż oba zadania zostały rozwiązane poprawnie. Co więcej, rozwiązania są bliskie rozwiązaniu wynikającemu z zastosowania algorytmu jacobianu pseudoodwrotnego. Widać, że w przypadku zadania 2 otrzymane rozwiązanie jest nieznacznie dalsze od rozwiązania jacobianu pseudoodwrotnego niż ma to miejsce w przypadku zadania 1. Także różnice otrzymanych prędkości średnich obu algorytmów w przypadku 2 są większe. Aby wytłumaczyć tę różnicę, na rysunku 4.42 przedstawione są dwie stronicie w przestrzeni konfiguracyjnej POLYCRANKa przechodzące przez liście początkowej E_α i końcowej E_β konfiguracji przegubowej, odpowiednio dla zadania pierwszego i drugiego. Widać, że trajektorie przegubowe otrzymane dzięki algorytmowi jacobianu rozszerzonego nie leżą na stronicach przechodzących przez E_α . W przypadku gdy trajektoria należałaby do liścia, aproksymacja byłaby idealna. Intuicyjnie możemy stwierdzić, że im trajektoria znajduje się bliżej liścia, tym lepszą otrzymujemy aproksymację. Biorąc pod uwagę zmienność Ω jako miarę „bliskości”, możemy obliczyć przyrost $\Delta\Omega = \Omega(x(0)) - \Omega(x(t_{\max}))$. Zakładając, że $t_{\max} = 10$, dla zadania pierwszego $\|\Delta\Omega\| = 0.490325$, natomiast dla zadania drugiego $\|\Delta\Omega\| = 0.755177$.

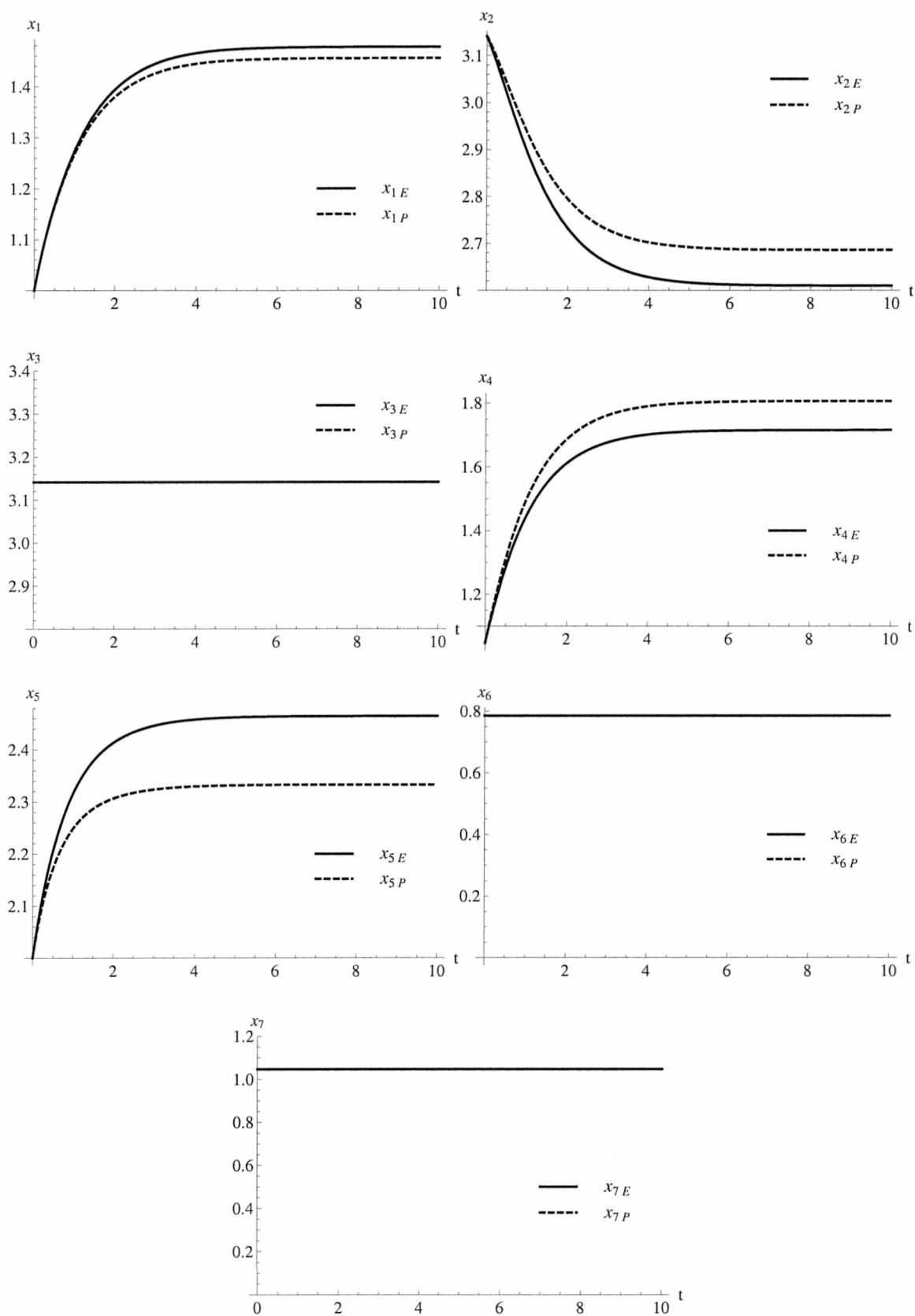
Wiadomo, że poprzez swoją postać algorytmy bazujące na jacobianie pseudoodwrotnym lokalnie minimalizują prędkości w przegubach w trakcie ruchu. Aby sprawdzić efektywność jacobianu aproksymującego, na rysunku 4.41 pokazano, że kwadraty normy prędkości przegubowych otrzymanych za pomocą obu algorytmów są sobie bliskie. Dodatkowo, w tabeli 4.10 zebrano wartości miar jakości aproksymacji (4.11) i (4.12) przy $t_{\max} = 10$.

Tabela. 4.10: Wartości miar jakości aproksymacji

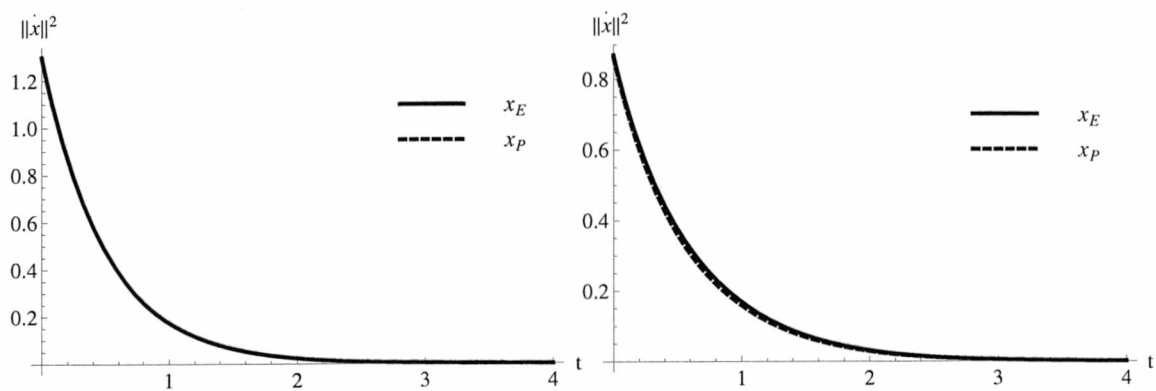
	Metoda geometryczna	
	δ_1	δ_2
Zadanie 1	0.00975	0.05157
Zadanie 2	0.02987	0.15749



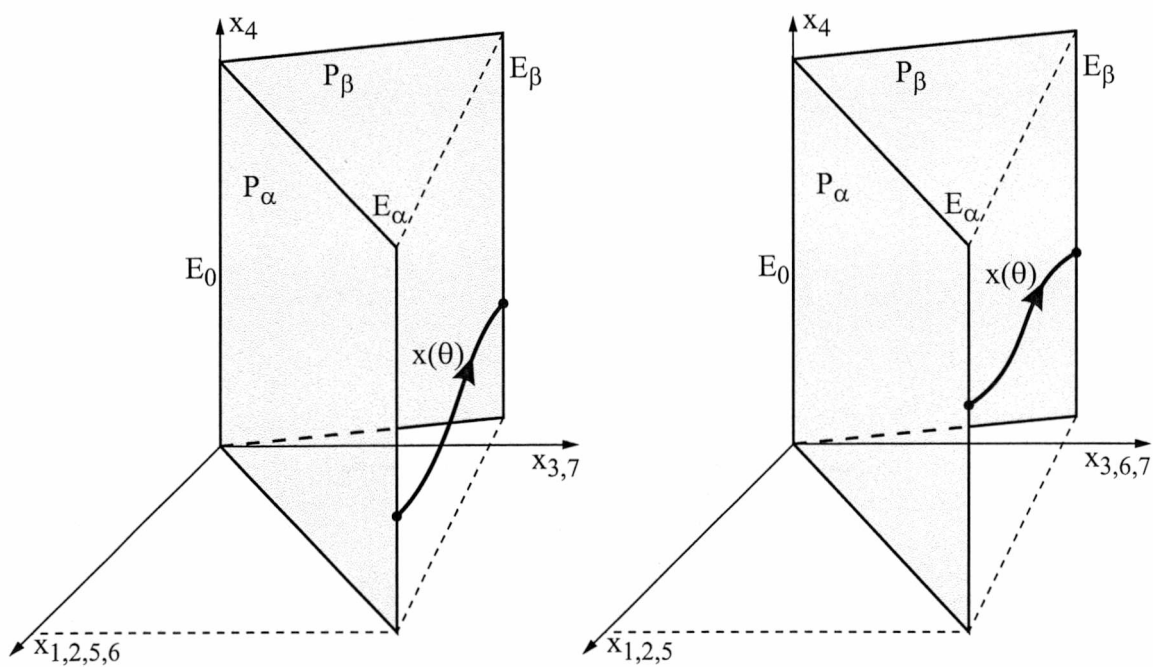
Rysunek 4.39: Rozwiązanie zadania 1



Rysunek 4.40: Rozwiązanie zadania 2



Rysunek 4.41: Kwadrat normy prędkości przegubowych dla zadania 1 (z lewej) i 2 (z prawej)



Rysunek 4.42: Stronice i trajektorie przegubowe dla zadania 1 (z lewej) i 2 (z prawej)

4.6 Porównanie metody geometrycznej i wariacyjnej

W niniejszym podrozdziale porównamy rozwiązania otrzymane dzięki metodzie geometrycznej i wariacyjnej z funkcjonalem błędu opisanym równaniem (2.37). Rozwiążemy trzy różne zadania kinematyki odwrotnej dla manipulatora POLYCRANK, każde za pomocą obu wspomnianych metod. Wyprowadzenie algorytmu metodą geometryczną i wszelkie założenia są takie same jak w podrozdziale 4.5.1. Dla metody wariacyjnej założyliśmy obszar całkowania równy

$$\mathbb{Q} = \left\{ x \in R^7 \left| 0.001 \leq x_1 \leq \frac{\pi}{2}, \frac{\pi}{2} + 0.001 \leq x_2 \leq \pi, 0 \leq x_3 \leq \pi, 0.001 \leq x_4 \leq \frac{\pi}{2}, \right. \right. \\ \left. \left. \frac{\pi}{2} + 0.001 \leq x_5 \leq \pi - 0.001, 0.001 \leq x_6 \leq \frac{\pi}{2}, 0 \leq x_7 \leq \pi \right\}.$$

Otrzymana za pomocą metody Ritza (2.60) optymalna, kwadratowa funkcja rozszerzająca przybiera postać

$$h(x) = 0.860096x_1 - 0.293629x_1^2 - 8.55278x_2 - 0.475787x_1x_2 + 0.485924x_2^2 \\ - 3.26389x_3 - 1.04843x_1x_3 + 1.73457x_2x_3 + 3.1366x_4 + 0.354132x_1x_4 \\ + 0.584715x_2x_4 + 11.1985x_5 + 1.34538x_1x_5 + 2.22106x_2x_5 \\ - 1.60976x_4x_5 - 4.09794x_5^2.$$

Dla wyznaczonej funkcji rozszerzającej budujemy odpowiedni jacobian rozszerzony oraz algorytm typu jacobianu rozszerzonego.

Rozwiązujemy trzy zadania, których odpowiednio punkty początkowe i zadane punkty końcowe w przestrzeni zadaniowej zebrane są w tabeli 4.11.

Tabela. 4.11: Dane potrzebne do obliczeń

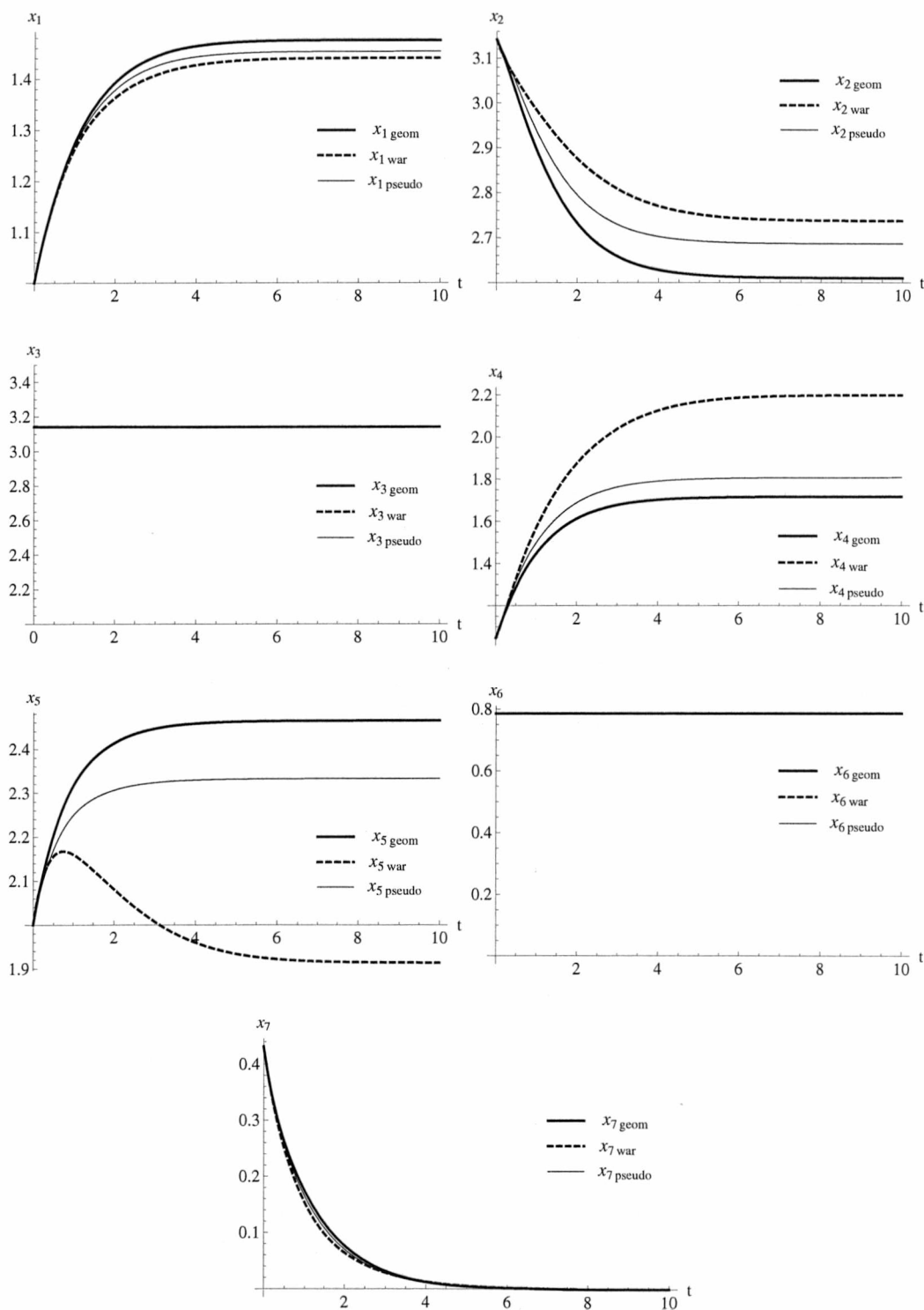
	Konfiguracja początkowa	Punkt końcowy
Zadanie 1	$x_0 = (1, \pi, \pi, \frac{\pi}{3}, 2, \frac{\pi}{4}, \frac{\pi}{3})$	$y_d = (-0.38, 0.64, 1.40, \pi, \frac{\pi}{4}, \frac{\pi}{3})$
Zadanie 2	$x_0 = (1, \pi, \pi, \frac{\pi}{6}, 2, \frac{\pi}{4}, \frac{\pi}{3})$	$y_d = (-0.38, 0.56, 1.54, \pi, \frac{\pi}{3}, \frac{\pi}{2})$
Zadanie 3	$x_0 = (0.2, \pi, \frac{\pi}{2}, 0.3, \frac{\pi}{2} + 0.001, \frac{\pi}{6}, \frac{\pi}{2})$	$y_d = (0.54, 0.07, 1.42, 0.1, \frac{\pi}{4}, \frac{\pi}{3})$

Rozwiązania powyższych zadań zostały pokazane na rysunkach 4.43, 4.44, 4.45. Rysunki 4.46, 4.47 i 4.48 przedstawiają normy prędkości w czasie $\|\dot{x}(t)\|^2$, natomiast rysunki 4.49, 4.50 i 4.51 obrazują przebiegi prędkości przegubowych $\dot{x}(t)$. W tabeli 4.12 zostały zebrane wartości miar jakości aproksymacji (4.11) i (4.12) przy $t_{\max} = 10$.

Tabela. 4.12: Wartości miar jakości aproksymacji

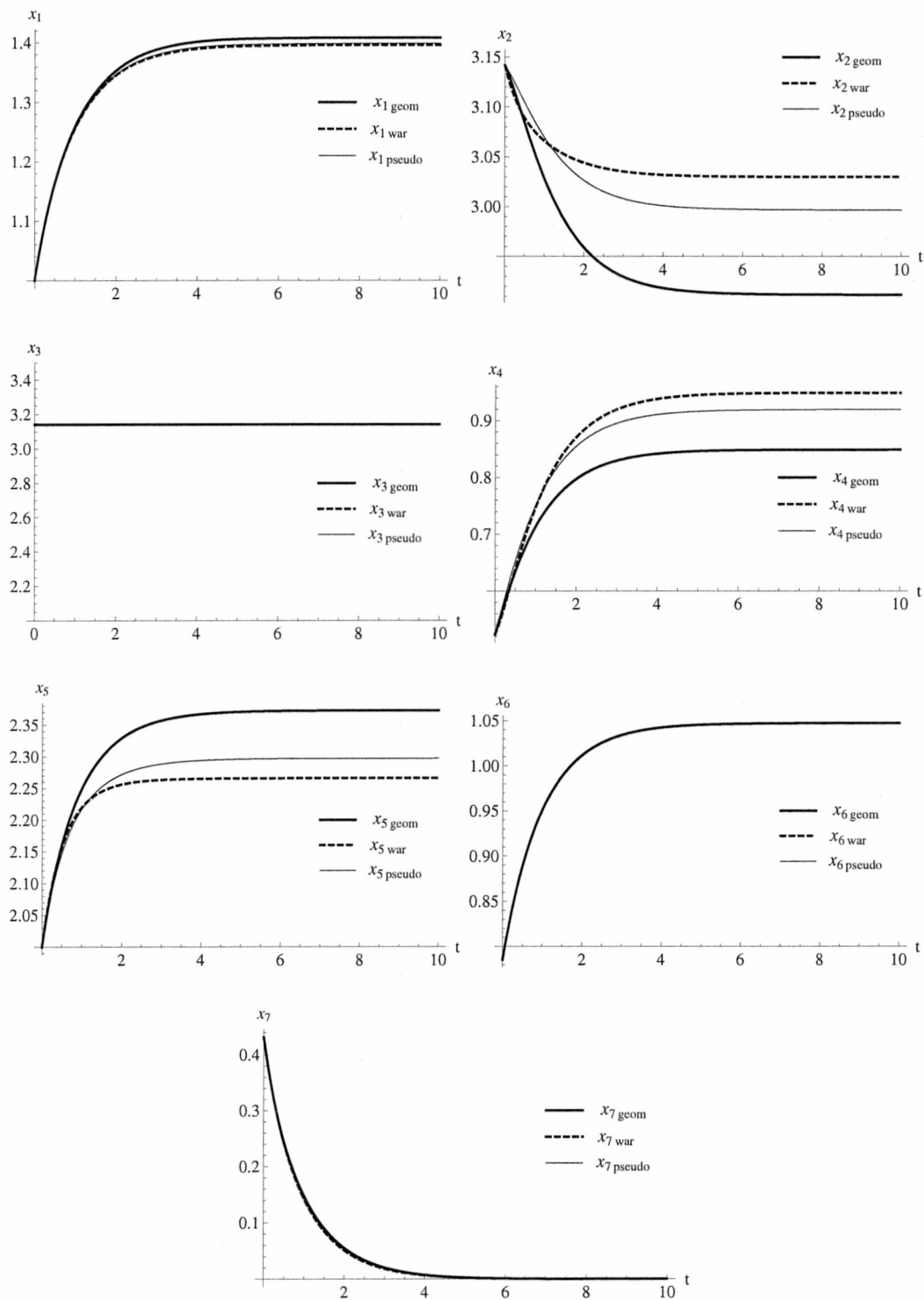
	Metoda wariacyjna		Metoda geometryczna	
	δ_1	δ_2	δ_1	δ_2
Zadanie 1	0.08859	0.42052	0.02987	0.15749
Zadanie 2	0.00847	0.11291	0.02202	0.12887
Zadanie 3	0.06308	0.69200	0.04123	0.90550

Analizując wyniki można zaobserwować, że zarówno algorytm uzyskany metodą geometryczną, jak i algorytm otrzymany za pomocą metody wariacyjnej zadowalająco rozwiązały rozważane zadania. Co więcej, otrzymane rozwiązania są bliskie rozwiązaniu otrzymanemu za pomocą algorytmu typu jakobianu pseudoodwrotnego. Jednakże, jakość aproksy-



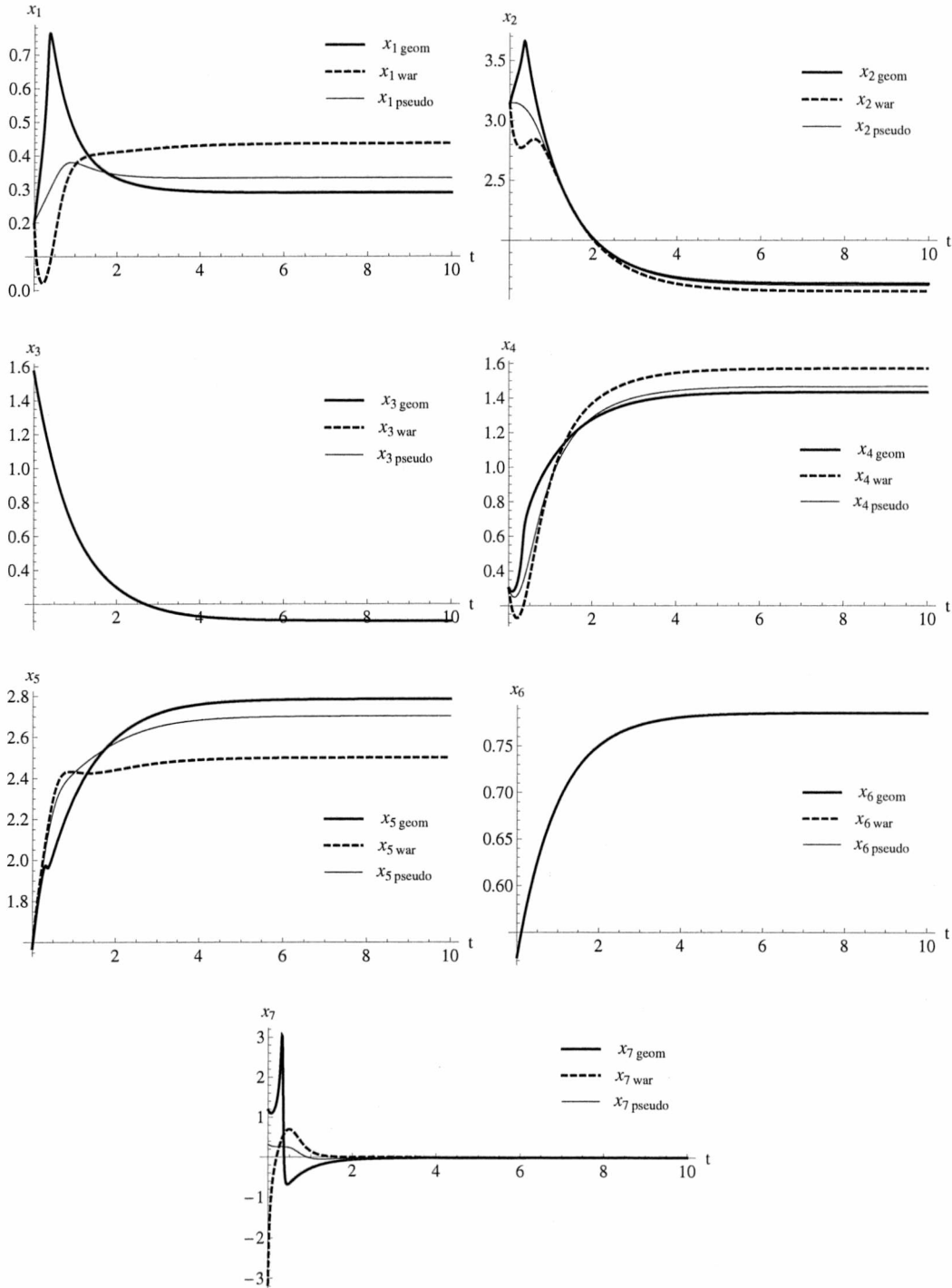
Rysunek 4.43: Rozwiązanie zadania 1

macji otrzymanej za pomocą metody geometrycznej zależy od zmienności 1-formy (4.19) w obrębie stronic w przestrzeni konfiguracyjnej. W metodzie geometrycznej, funkcja rozszerzająca obliczana jest numerycznie tylko w konfiguracjach wygenerowanych przez algorytm kinematyki odwrotnej. Inaczej jest w metodzie wariacyjnej, gdzie uzyskuje się

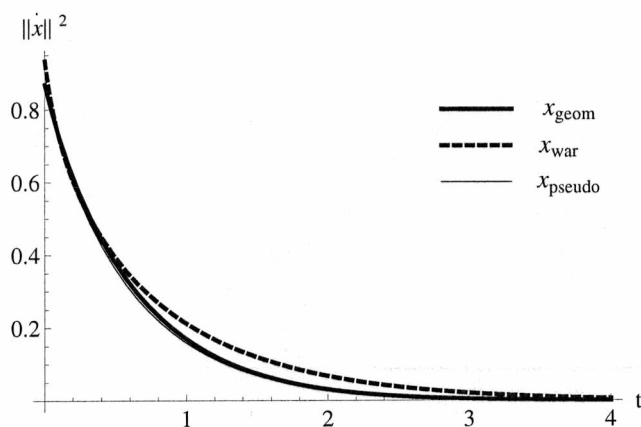


Rysunek 4.44: Rozwiązanie zadania 2

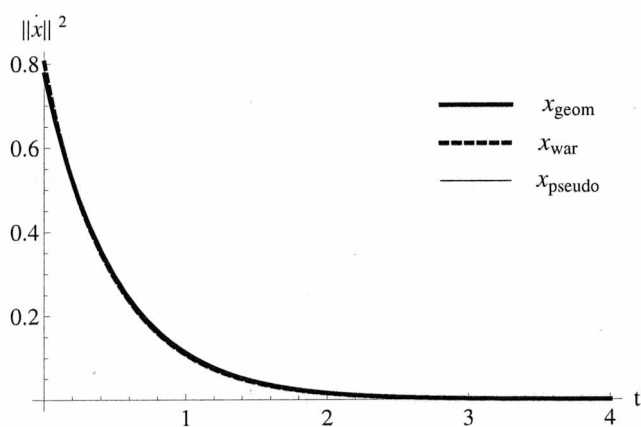
analityczną postać funkcji rozszerzającej na pewnym obszarze przestrzeni konfiguracyjnej. Zaletą stosowania metody geometrycznej jest stosunkowo krótki czas obliczeń, z drugiej jednak strony, funkcja rozszerzająca, jest zależna od zadania. Z tego powodu, musi być ona wyliczana każdorazowo dla kolejnego zadania. Obliczenia dla metody wariacyjnej są bardziej czasochłonne, lecz raz wyznaczona optymalna funkcja rozszerzająca pozwala rozwiązać wiele zadań, niezależnie od ich postaci.



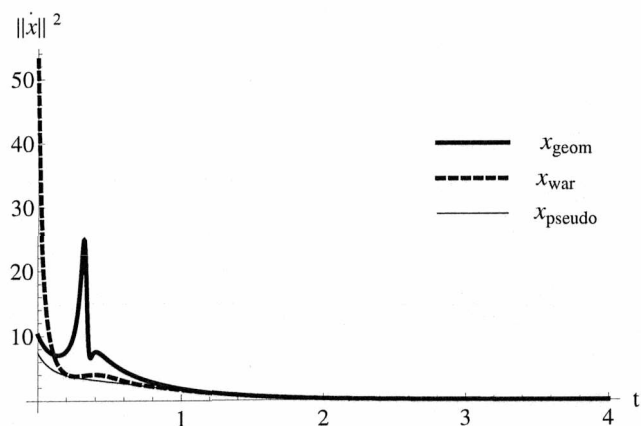
Rysunek 4.45: Rozwiązanie zadania 3



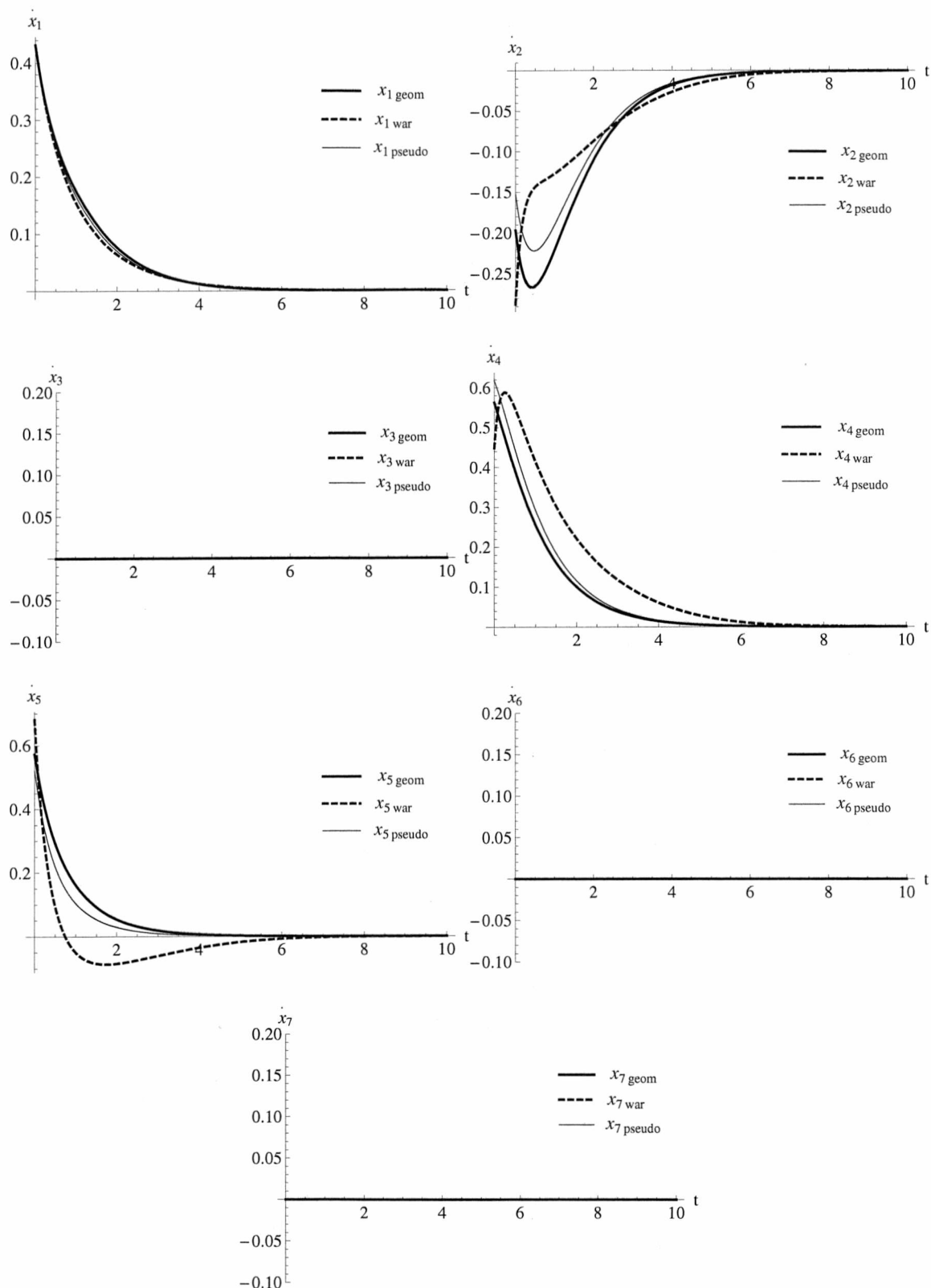
Rysunek 4.46: Normy prędkości przegubowych w zadaniu 1



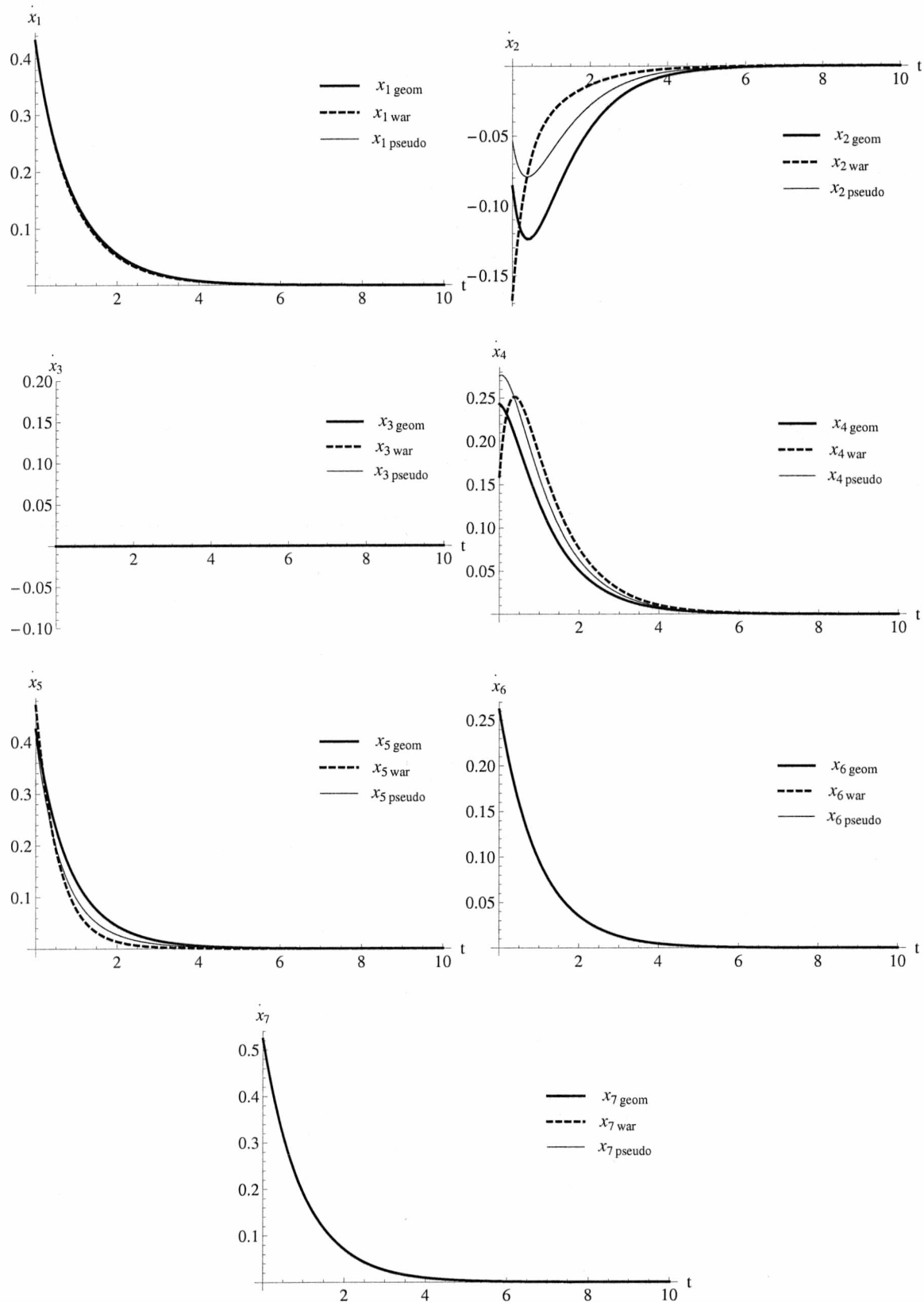
Rysunek 4.47: Normy prędkości przegubowych w zadaniu 2



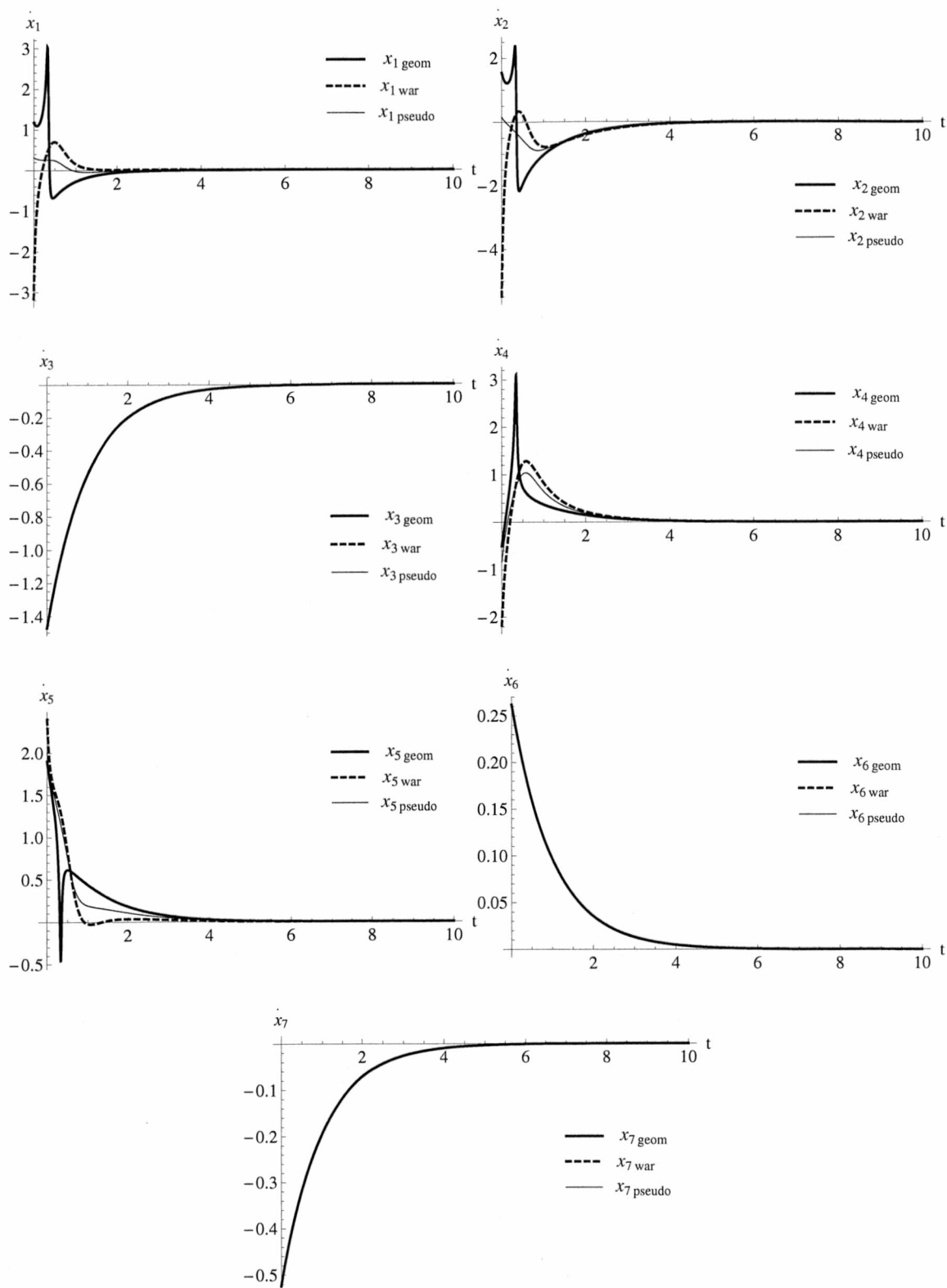
Rysunek 4.48: Normy prędkości przegubowych w zadaniu 3



Rysunek 4.49: Prędkości przegubowe otrzymane dla zadania 1



Rysunek 4.50: Prędkości przegubowe otrzymane dla zadania 2



Rysunek 4.51: Prędkości przegubowe otrzymane dla zadania 3

4.7 Aspekty obliczeniowe

Przytoczmy kilka spostrzeżeń związanych z implementacją metod aproksymacji algorytmów jacobianowych. Wszystkie obliczenia numeryczne zostały przeprowadzone w programie MATHEMATICA.

W obliczeniach odnoszących się do metody Ritza, postać funkcji rozszerzającej była wybierana w klasie funkcji liniowych, kwadratowych i 3-stopnia. Uzyskane wyniki wskazują na to, iż jakość aproksymacji stosunkowo nieznacznie wzrasta wraz ze wzrostem stopnia wielomianu. Z drugiej strony, zwiększenie stopnia wielomianu wiąże się z istotnie większym nakładem obliczeniowym. Dla manipulatora POLYCRANK (o $n = 7$ stopniach swobody) liczba współczynników $a \in \mathbb{R}^p$ dla liniowej funkcji rozszerzającej jest równa $p = 7$, dla kwadratowej $p = 35$, a w przypadku funkcji wielomianowej trzeciego stopnia wynosi $p = 120$. Zgodnie z (2.55), (2.59), wyznaczenie optymalnych wartości współczynników polega na obliczeniu macierzy wymiaru $p \times p$ i wektora $p \times 1$, których elementami są całki n -krotne. W związku z tym, czas obliczeń w przypadku funkcji liniowej jest nawet piętnastokrotnie krótszy niż dla funkcji kwadratowej. Reasumując, biorąc pod uwagę nieznaczną poprawę jakości aproksymacji wraz ze wzrostem stopnia wielomianu i znaczny wzrost czasu obliczeń w zależności od p jest racjonalne, żeby w tych przypadkach, reprezentować funkcję rozszerzającą za pomocą wielomianu pierwszego stopnia.

Przeprowadzone badania porównawcze aproksymacji algorytmów jacobianowych dla kinematyki w $SE(3)$ i dla kinematyki wyrażonej we współrzędnych wykazują, że z punktu widzenia jakości aproksymacji obie reprezentacje kinematyki dają zbliżone wyniki. Jednakże, czas potrzebny na otrzymanie optymalnej funkcji rozszerzającej w przypadku kinematyki w $SE(3)$ jest około dwa razy krótszy niż w przypadku kinematyki we współrzędnych. Daje to preferencje reprezentacji kinematyki bez współrzędnych.

Porównując efektywność aproksymacji otrzymanej metodą geometryczną i metodą wariacyjną można dojść do wniosku, że obie metody dają zadowalające wyniki. Jednakże, jakość aproksymacji uzyskanej metodą geometryczną zależy od zmienności 1-formy w obrębie stronic w przestrzeni stanu. W metodzie geometrycznej, funkcja rozszerzająca obliczana jest numerycznie tylko w konfiguracjach wygenerowanych przez algorytm kinematyki odwrotnej. Inaczej jest w metodzie wariacyjnej, gdzie uzyskuje się analityczną postać funkcji rozszerzającej na pewnym obszarze przestrzeni konfiguracyjnej. Badania wykazały, że w przypadku manipulatora POLYCRANK, czas potrzebny na obliczenia metodą wariacyjną był od pięciu razy (dla liniowej funkcji rozszerzającej) do sześćdziesięciu razy (dla kwadratowej funkcji rozszerzającej) dłuższy niż dla metody geometrycznej. Zalecane zastosowanie metody geometrycznej jest stosunkowo krótki czas obliczeń, ale w tym przypadku funkcja rozszerzająca musi być wyliczana dla każdego zadania indywidualnie. Wariacyjne zadanie optymalizacji jest bardziej czasochłonne obliczeniowo, jednak otrzymana funkcja rozszerzająca może być zastosowana do wielu różnych zadań, niezależnie od punktu początkowego i docelowego.

Rozdział 5

Przykłady numeryczne — zadanie reprodukcji trajektorii

5.1 Zadanie reprodukcji trajektorii stanu

W tym podrozdziale przedstawimy wyniki obliczeń numerycznych dla afinicznych układów sterowania. Dla uproszczenia przyjmiemy założenie $B^T(t)B(t) = I_m$, które jest spełnione w modelach użytych do obliczeń, ale także, na przykład, w przypadku samochodu kinematycznego. W rozdziale zawarte są przykłady reprodukcji trajektorii stanu i wyjścia. Obliczenia numeryczne będziemy przeprowadzać na modelach podwójnego integratora, kinematyki monocykla, dynamiki łyżwiarza Czapłygina i kuli kinematycznej.

5.1.1 Podwójny integrator

Zacznijmy od bardzo prostego przykładu. Podwójny integrator jest układem liniowym, opisanym równaniem

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u = Ax + Bu,$$

co oznacza, że niezależnie od θ i t , $A_\theta(t) = A$ i $B_\theta(t) = B$. Korzystając z tej obserwacji i z faktu, że $B^T B = 1$, uzyskujemy równanie reprodukcji (3.13) w następującej postaci [55]

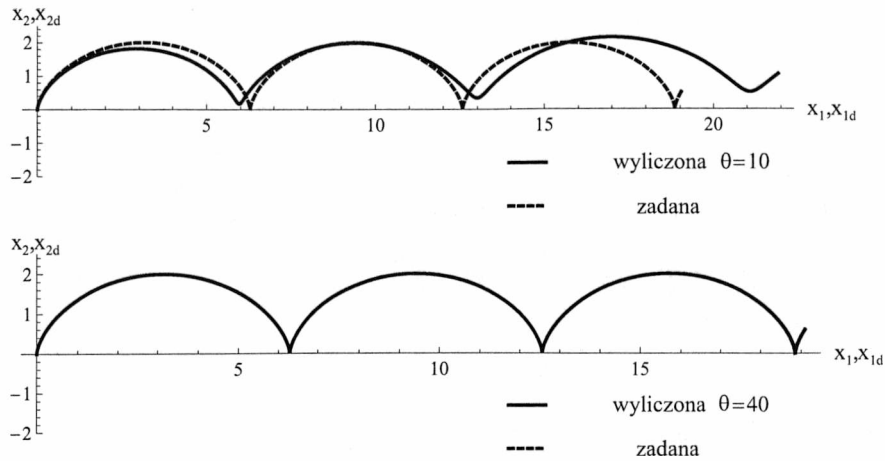
$$\frac{\partial^2 u(t, \theta)}{\partial t \partial \theta} + \alpha \frac{\partial u(t, \theta)}{\partial \theta} + \gamma \frac{\partial u(t, \theta)}{\partial t} + \alpha \gamma u(t, \theta) = \gamma (\ddot{x}_{d2}(t) + \alpha \dot{x}_{d2}(t)),$$

z warunkami brzegowymi $u(t, 0) = u_0(t)$ i $u(0, \theta)$ będącym rozwiązaniem równania

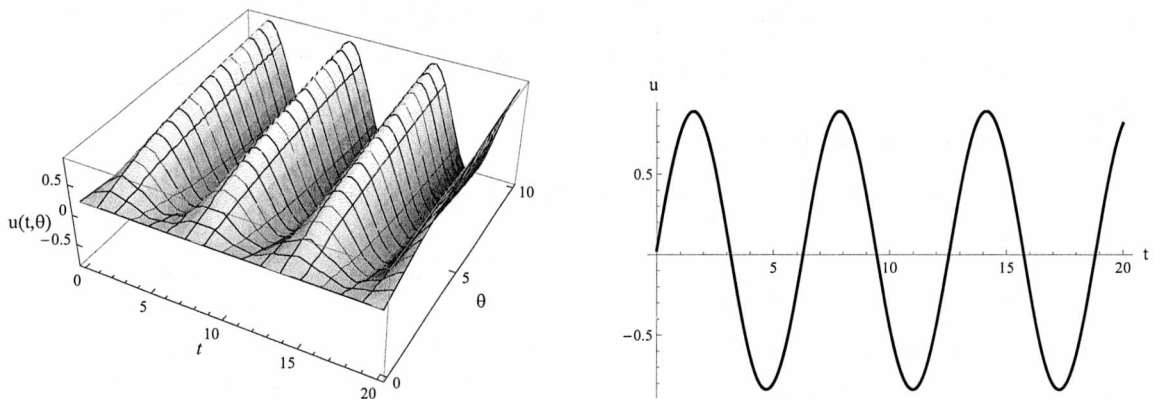
$$\frac{\partial u(0, \theta)}{\partial \theta} + \gamma u(0, \theta) = \gamma \dot{x}_{d2}(0).$$

Rozwiązanie powyższego zagadnienia brzegowego prowadzi do wyznaczenia funkcji sterującej, pod wpływem której jest realizowana zadana trajektoria. Zgodnie z ideą metody homotopii, rozwiązanie zadania oryginalnego otrzymuje się dla $\theta \rightarrow +\infty$. Dla przykładu, wybierzmy trajektorię zadaną w postaci cykloidy $x_{d1}(t) = t - \sin t$, $x_{d2}(t) = 1 - \cos t$. Naszym zadaniem jest wyznaczenie sterowania reprodukcją tej trajektorii. W implementacji numerycznej współczynniki w funkcji homotopii wynoszą odpowiednio: $\alpha = 1$, $\gamma = 0.2$, czas obliczeń $t_{\max} = 20$, warunek początkowy $x(0) = (0, 0)$, sterowanie początkowe $u(t, 0) = 0.2$. Przeprowadzono obliczenia dla różnych wartości współczynnika θ . Otrzymane wyniki są zgodne z oczekiwaniem, że dla większych wartości θ , uzyskana

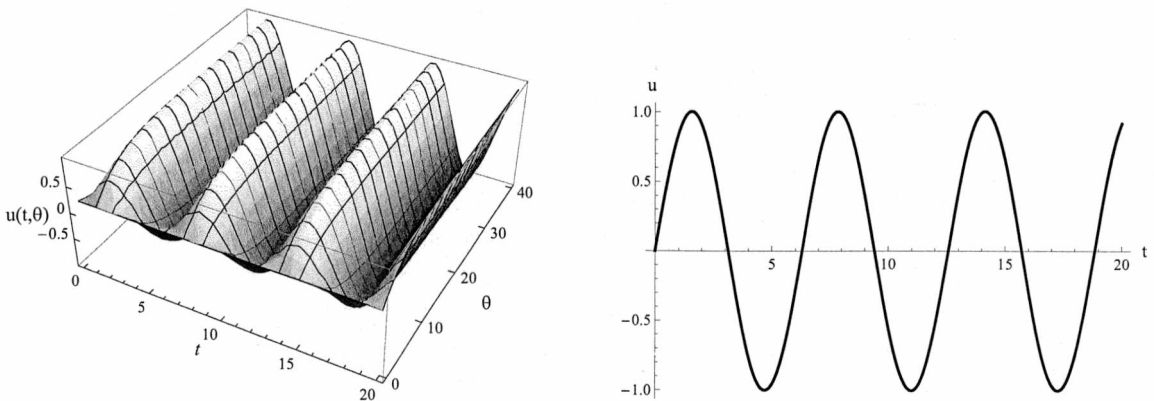
trajektoria ruchu jest bliższa zadanej. Na rysunku 5.1 przedstawiono ścieżkę wyliczoną i zadaną dla dwóch różnych wartości θ , natomiast na rysunkach 5.2 i 5.3 pokazano rozwiązanie równania reprodukcji i otrzymaną funkcję sterującą.



Rysunek 5.1: Wyliczona i zadana ścieżka ruchu dla podwójnego integratora przy $\theta = 10$ i $\theta = 40$



Rysunek 5.2: Rozwiązanie równania reprodukcji i otrzymane sterowanie dla $\theta = 10$



Rysunek 5.3: Rozwiązanie równania reprodukcji i otrzymane sterowanie dla $\theta = 40$

5.1.2 Monocykl

Kolejnym przykładem, dla którego rozwiążemy zadanie reprodukcji trajektorii stanu jest model kinematyki monocykla poruszającego się bez poślizgu bocznego. Model monocykla pojawił się w podrozdziale 4.4.2, dla kompletności, przypomnijmy równanie kinematyki

$$\begin{cases} \dot{x}_1 = u_1 \cos x_3 \\ \dot{x}_2 = u_1 \sin x_3 \\ \dot{x}_3 = u_2, \end{cases}$$

gdzie x_1, x_2 są współrzędnymi położenia, x_3 jest orientacją robota, natomiast sterowania u_1 i u_2 oznaczają odpowiednio prędkość wzdłużną i kątową.

Teraz, przy zadanym $(u(t), x(t))$, wyznaczmy macierze przybliżenia liniowego dla powyższego modelu

$$A(t) = \begin{bmatrix} 0 & 0 & -u_1(t) \sin x_3(t) \\ 0 & 0 & u_1(t) \cos x_3(t) \\ 0 & 0 & 0 \end{bmatrix}$$

oraz

$$B(t) = \begin{bmatrix} \cos x_3(t) & 0 \\ \sin x_3(t) & 0 \\ 0 & 1 \end{bmatrix}.$$

Jak łatwo zauważyć, warunek $B^T(t)B(t) = I_2$ jest spełniony. Korzystając z niego oraz po wyliczeniu

$$\begin{aligned} \dot{A}(t) &= \begin{bmatrix} 0 & 0 & -\dot{u}_1(t) \sin x_3(t) - u_1(t)u_2(t) \cos x_3(t) \\ 0 & 0 & \dot{u}_1(t) \cos x_3(t) - u_1(t)u_2(t) \sin x_3(t) \\ 0 & 0 & 0 \end{bmatrix}, \\ \dot{B}(t) &= \begin{bmatrix} -u_2(t) \sin x_3(t) & 0 \\ u_2(t) \cos x_3(t) & 0 \\ 0 & 0 \end{bmatrix} \end{aligned}$$

i dokonaniu odpowiednich podstawień w (3.13), otrzymujemy równanie reprodukcji trajektorii monocykla [55]

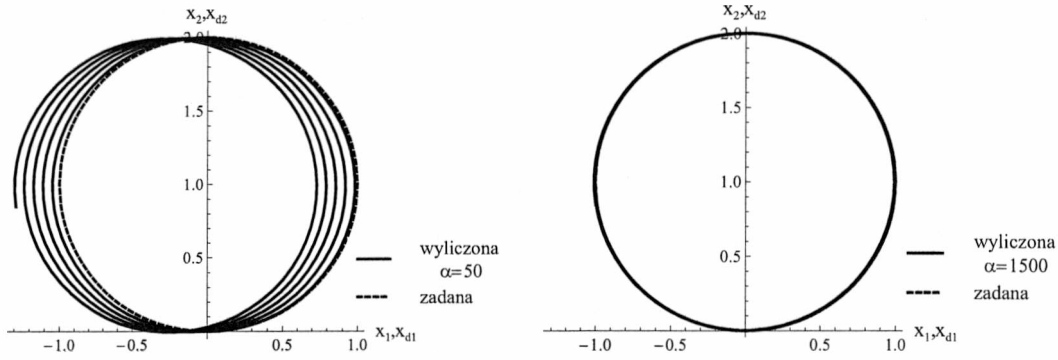
$$\begin{aligned} \frac{\partial^2 u(t, \theta)}{\partial t \partial \theta} + \begin{bmatrix} \alpha & \alpha^{-1} u_1(t, \theta) u_2(t, \theta) \\ 0 & \alpha \end{bmatrix} \frac{\partial u(t, \theta)}{\partial \theta} + \gamma \frac{\partial u(t, \theta)}{\partial t} \\ + \gamma \begin{bmatrix} \alpha & \alpha^{-1} u_1(t, \theta) u_2(t, \theta) \\ 0 & \alpha \end{bmatrix} u(t, \theta) = \gamma \begin{bmatrix} \cos x_3(t, \theta) & \sin x_3(t, \theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ \times (\ddot{x}_d(t) + \alpha \dot{x}_d(t)) + \gamma \begin{bmatrix} u_1(t, \theta) u_2(t, \theta) (\dot{x}_{d3} - (x_3(t, \theta) - x_{d3}(t))) \\ 0 \end{bmatrix}. \end{aligned}$$

Warunek brzegowy $u(t, 0) = u_0(t)$, a warunek początkowy $u(0, \theta)$ jest rozwiązaniem liniowego równania różniczkowego

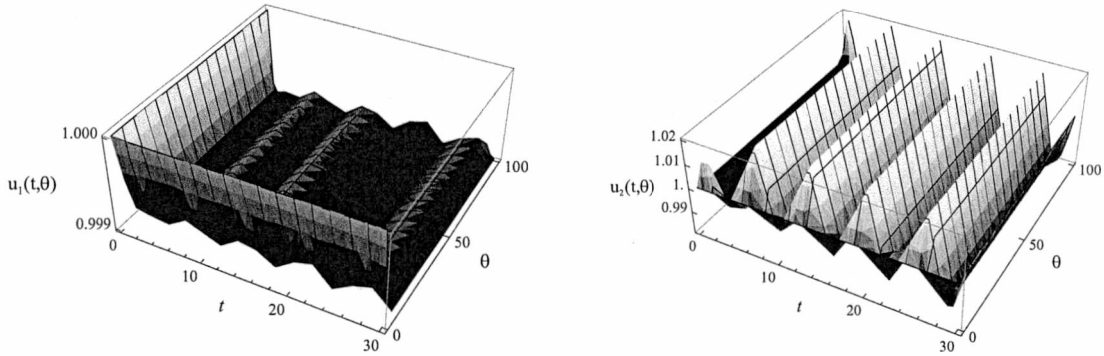
$$\frac{\partial u(0, \theta)}{\partial \theta} + \gamma u(0, \theta) = \gamma \begin{bmatrix} \cos x_{d3}(0) & \sin x_{d3}(0) & 0 \\ 0 & 0 & 1 \end{bmatrix} \dot{x}_d(0).$$

Podobnie jak w poprzednim przykładzie, rozwiązujemy zadanie reprodukcji z warunkiem brzegowym, dla trajektorii $x_{d1}(t) = \sin t$, $x_{d2} = 1 - \cos t$, $x_{d3}(t) = t$ definiującej okrąg

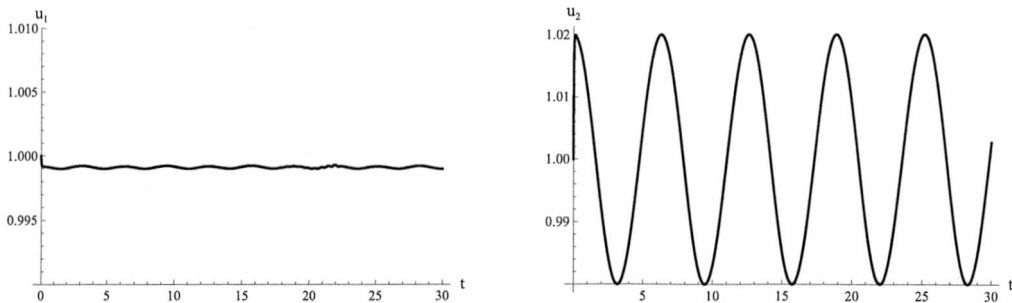
o środku w punkcie $(x_1, x_2) = (0, 1)$ i promieniu 1. W poprzednim przykładzie pokazaliśmy wpływ współczynnika θ na jakość otrzymywanych wyników. Dla kompletności, przedstawimy wyniki obliczeń dla różnych wartości współczynnika α . Pozostałe parametry są następujące: $\gamma = 1$, czas symulacji $t_{\max} = 30$, $\theta = 100$, warunek początkowy $x_1(0) = 0$, $x_2(0) = 0$, $x_3(0) = 0$ oraz $u_1(t, 0) = u_1(t, 0) = 1$. Na rysunku 5.4 pokazaliśmy wyliczoną i zadaną ścieżkę robota na płaszczyźnie dla dwóch wartości parametru $\alpha = 50$ i $\alpha = 1500$. Uzyskane wyniki pokazują, że wzrost α zwiększa prędkość zbieżności wyliczonej ścieżki ruchu do ścieżki zadanej. Rozwiązanie równania reprodukcji oraz otrzymane sterowanie dla różnych α przedstawiają rysunki 5.5 – 5.8.



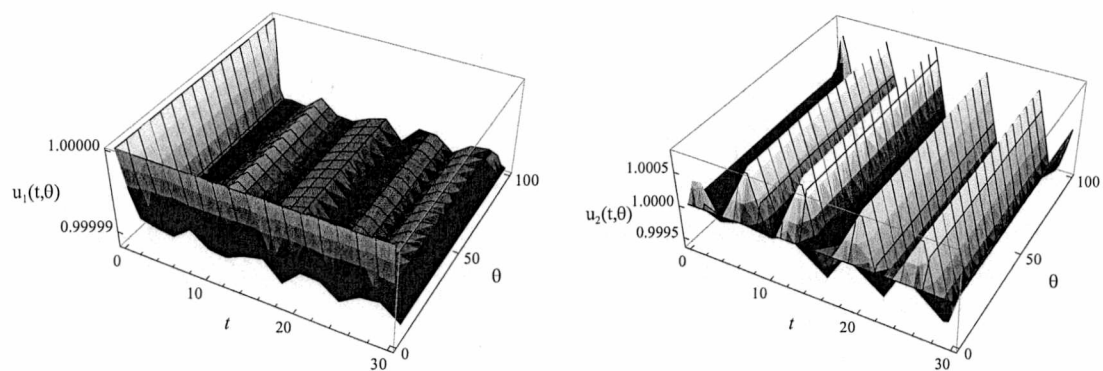
Rysunek 5.4: Wyliczona i zadana ścieżka ruchu dla monocykla przy $\alpha = 50$ i $\theta = 1500$



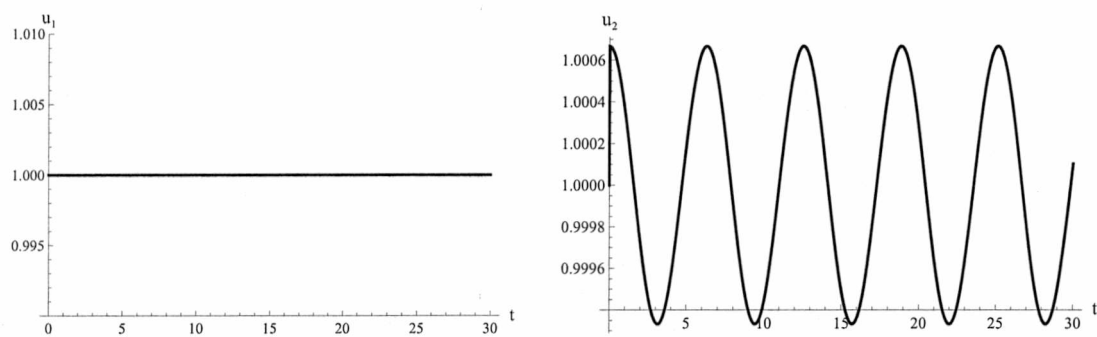
Rysunek 5.5: Rozwiązanie równania reprodukcji dla monocykla przy $\alpha = 50$



Rysunek 5.6: Otrzymane sterowanie dla monocykla przy $\alpha = 50$



Rysunek 5.7: Rozwiązanie równania reprodukcji dla monocykla przy $\alpha = 1500$



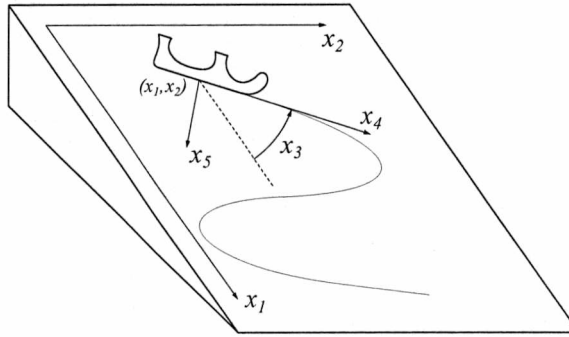
Rysunek 5.8: Otrzymane sterowanie dla monocykla przy $\alpha = 1500$

5.1.3 Łyżwiarz Czapłygina

Poniższy przykład obrazuje rozwiązanie zadania reprodukcji trajektorii stanu dla układu z dynamiką. Rysunek 5.9 przedstawia łyżwiarza Czapłygina [8], którego równanie dynamiki jest określone przez afiniczny układ sterowania z dryfem

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{pmatrix} = \begin{bmatrix} x_4 \cos x_3 \\ x_4 \sin x_3 \\ x_5 \\ \cos x_3 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = f(x) + G(x)u. \quad (5.1)$$

W układzie (5.1) współrzędne x_1, x_2, x_3 oznaczają odpowiednio położenie oraz orientację łyżwiarza, x_4 jest prędkością liniową a x_5 określa prędkość kątową, u_1 i u_2 to siły zewnętrzne działające na łyżwiarza.



Rysunek 5.9: Łyżwiarz Czapłygina

Macierze przybliżenia liniowego wzdłuż pary sterowanie–trajektoria $(u(t), x(t))$ są następujące

$$A(t) = \begin{bmatrix} 0 & 0 & -x_4(t) \sin x_3(t) & \cos x_3(t) & 0 \\ 0 & 0 & x_4(t) \cos x_3(t) & \sin x_3(t) & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -\sin x_3(t) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad B(t) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

oraz odpowiednie pochodne względem czasu t

$$\begin{aligned} \dot{A}(t) &= \begin{bmatrix} 0 & 0 & -\dot{x}_4(t) \sin x_3(t) - x_4(t) \cos x_3(t) \dot{x}_3(t) & -\sin x_3(t) \dot{x}_3(t) & 0 \\ 0 & 0 & \dot{x}_4(t) \cos x_3(t) - x_4(t) \sin x_3(t) \dot{x}_3(t) & \cos x_3(t) \dot{x}_3(t) & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\cos x_3(t) \dot{x}_3(t) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \\ \dot{B}(t) &= [0]_{5 \times 2}. \end{aligned}$$

Korzystając z własności $B^T(t)B(t) = I_2$ oraz korzystając z powyższych obliczeń, równanie

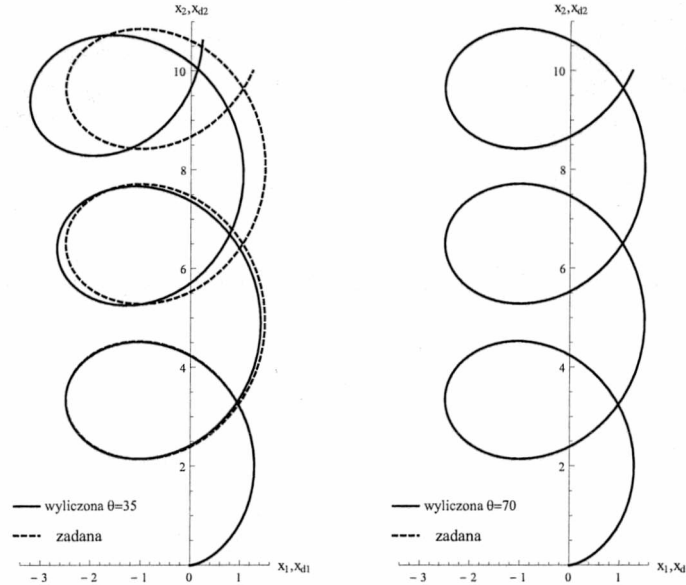
reprodukcji trajektorii dla łyżwiarza Czapłygina przyjmuje następującą postać

$$\begin{aligned} & \frac{\partial^2 u(t, \theta)}{\partial t \partial \theta} + \begin{bmatrix} \alpha & -\alpha^{-2} \cos(x_3) \dot{x}_3 \\ 0 & \alpha \end{bmatrix} \frac{\partial u(t, \theta)}{\partial \theta} + \gamma \frac{\partial u(t, \theta)}{\partial t} + \gamma \alpha \begin{bmatrix} 1 & -\alpha^{-3} \cos(x_3) \dot{x}_3 \\ 0 & 1 \end{bmatrix} u(t, \theta) = \\ & \gamma \begin{bmatrix} \ddot{x}_{4d} \\ \ddot{x}_{5d} \end{bmatrix} - \begin{bmatrix} 0 & 0 & \gamma(-\alpha^{-1} \cos(x_3) \dot{x}_3 - \sin(x_3)) & -\alpha \gamma & \gamma \alpha^{-2} \cos(x_3) \dot{x}_3 \\ 0 & 0 & 0 & 0 & -\alpha \gamma \end{bmatrix} \dot{x}_d(t) \\ & + \gamma \alpha \begin{bmatrix} 0 & 0 & -\alpha^{-1} \cos(x_3) \dot{x}_3 - \sin(x_3) & 0 & \alpha^{-2} \cos(x_3) \dot{x}_3 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} e(t, \theta) \\ & + \begin{bmatrix} \gamma(-\alpha \cos(x_3) - \alpha^{-1} \cos(x_3) \dot{x}_3 x_5) \\ 0 \end{bmatrix} \end{aligned} \quad (5.2)$$

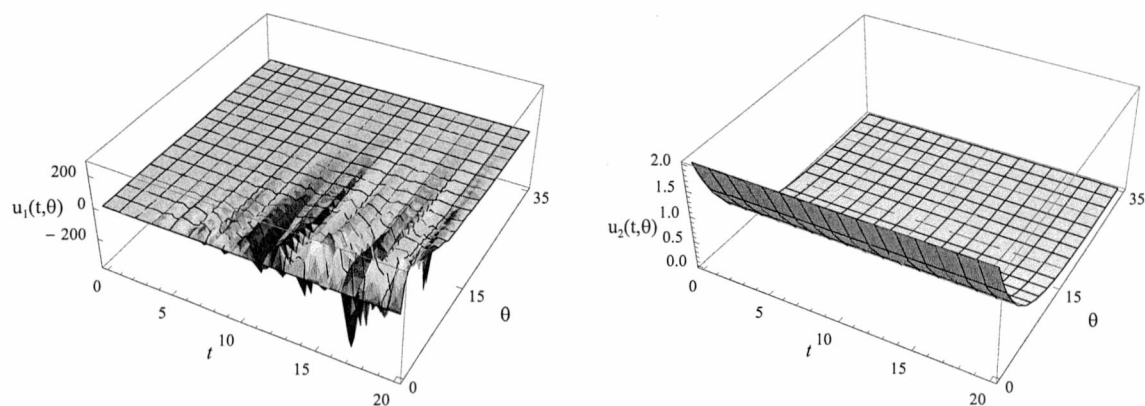
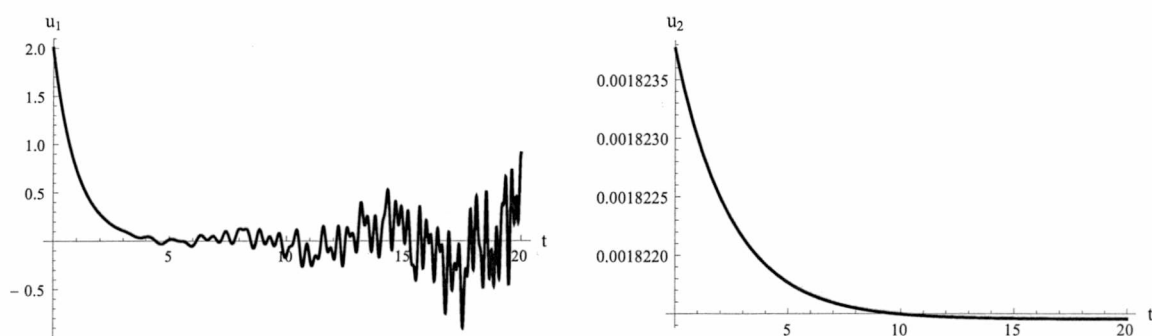
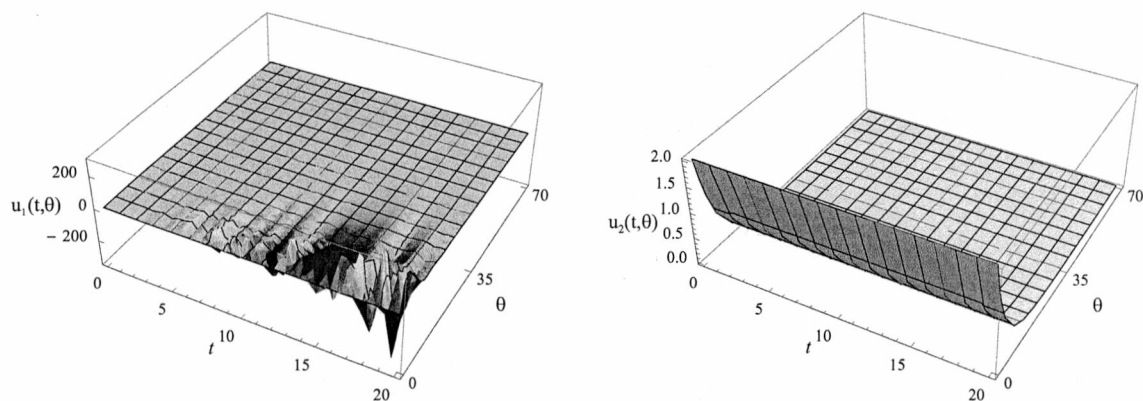
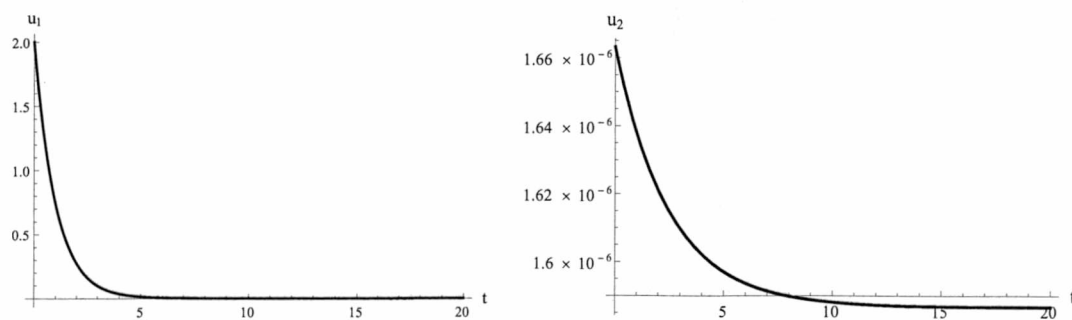
z następującym warunkiem brzegowym $u(t, 0) = u_0(t)$ oraz $u(0, \theta)$ będącym rozwiązaniem

$$\frac{\partial u(0, \theta)}{\partial \theta} + \gamma u(0, \theta) = \gamma \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \left(\dot{x}_d(0) - \begin{bmatrix} x_4(0) \cos x_3(0) \\ x_4(0) \sin x_3(0) \\ x_5(0) \\ \cos x_3(0) \\ 0 \end{bmatrix} \right). \quad (5.3)$$

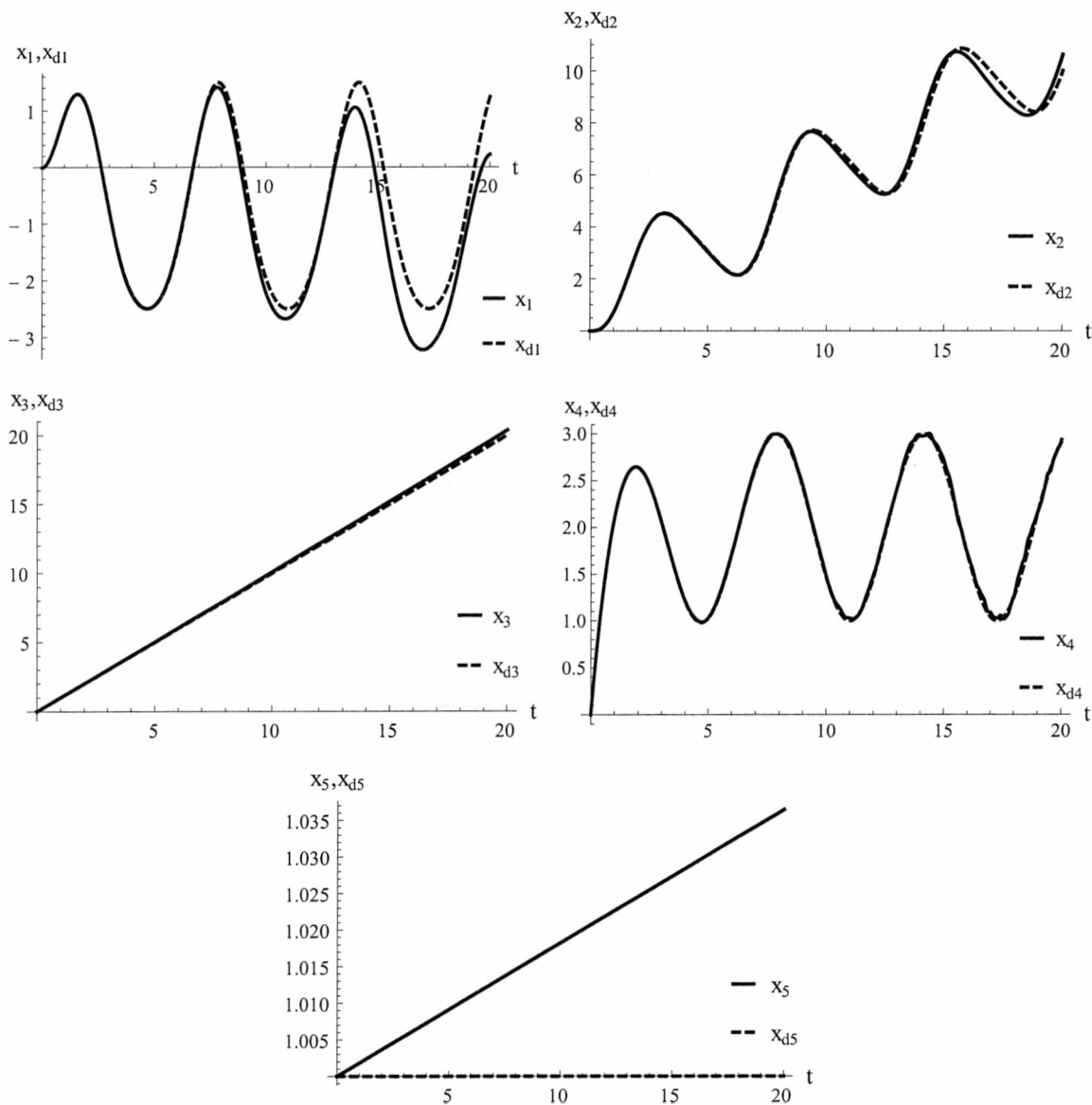
Rozwiązując równanie reprodukcji (5.2) z warunkiem brzegowym (5.3), przy $\theta \rightarrow +\infty$ otrzymamy funkcję sterującą, która będzie generowała zadaną trajektorię. Niech trajektoria zadana określona będzie jako $x_{d1} = \exp(-t)(\cos t - \sin t) + \frac{1}{4}(-3 - \cos(2t) + 8 \sin t)$, $x_{d2}(t) = \exp(-t)(\cos t + \sin t) + \frac{1}{2}(2 + t - \cos t(4 + \sin t))$, $x_{d3}(t) = t$, $x_{d4}(t) = \exp(-t)(-2 + 2 \exp t + \exp t \sin t)$, $x_{d5}(t) = 1$. Pozostałe parametry są równe $\alpha = 0.4$, $\gamma = 0.2$, czas symulacji $t_{\max} = 20$, warunki początkowe $x(0) = (0, 0, 0, 0, 1)$ oraz $u(t, 0) = (2, 2)$. Obliczenia wykonano dla dwóch wartości $\theta = 35$ oraz $\theta = 70$. Rysunek 5.10 przedstawia trajektorie łyżwiarza dla dwóch wartości θ . Rysunki 5.11, 5.13 ilustrują rozwiązanie równania reprodukcji, natomiast 5.12 i 5.14 pokazują otrzymane funkcje sterujące. Przebiegi poszczególnych zmiennych stanu w czasie zostały ukazane na rysunkach 5.15 oraz 5.16.



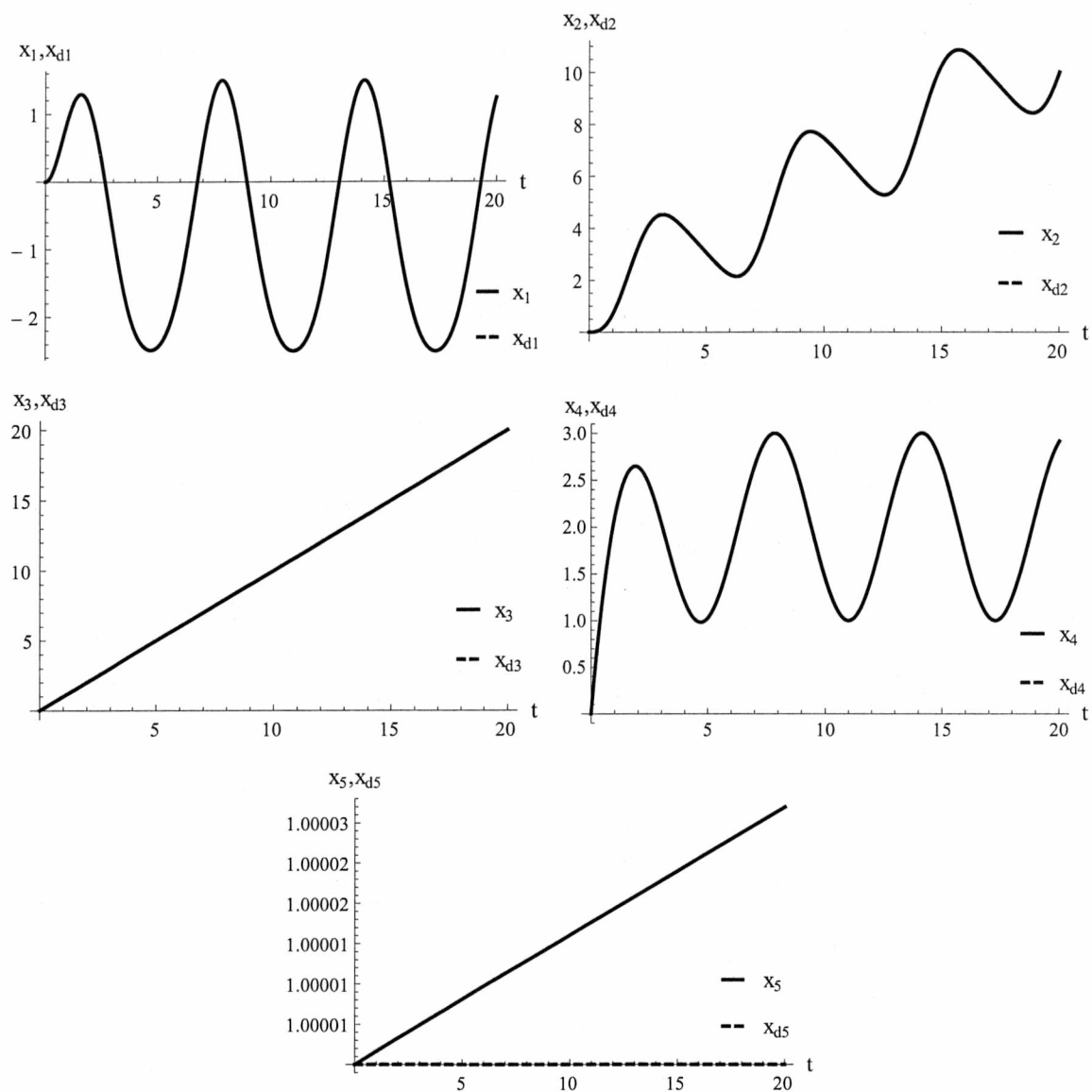
Rysunek 5.10: Trajektoria łyżwiarza Czapłygina dla $\theta = 35$ (z lewej) oraz dla $\theta = 70$ (z prawej)

Rysunek 5.11: Rozwiązanie równania reprodukcji trajektorii dla $\theta = 35$ Rysunek 5.12: Otrzymane funkcje sterujące dla $\theta = 35$ Rysunek 5.13: Rozwiązanie równania reprodukcji trajektorii dla $\theta = 70$ Rysunek 5.14: Otrzymane funkcje sterujące dla $\theta = 70$

Analizując otrzymane wyniki możemy stwierdzić, że już przy niewielkich wartościach parametru $\theta = 70$ zadanie reprodukcji trajektorii stanu łyżwiarza Czapłygina zostało rozwiązane poprawnie. Wartość parametru, dla którego otrzymane funkcje sterujące będą zwracały trajektorię wystarczająco bliską trajektorii zadanej, zależy od modelu, dla którego rozwiązujemy zadanie reprodukcji trajektorii.



Rysunek 5.15: Przebiegi zmiennych stanu w czasie dla łyżwiarza przy $\theta = 35$



Rysunek 5.16: Przebiegi zmiennych stanu w czasie dla łyżwiarza przy $\theta = 70$

5.2 Zadanie reprodukcji trajektorii wyjścia

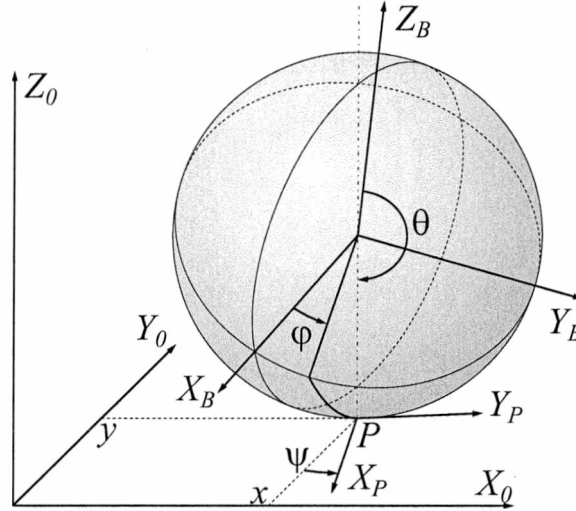
Jako ilustrację działania algorytmu reprodukcji trajektorii dla afinicznego układu sterowania z wyjściem przedstawimy wyniki obliczeń numerycznych dla kuli kinematycznej.

5.2.1 Kula kinematyczna

Kinematyka kuli toczącej się po płaszczyźnie wyprowadzona w [8, 28] jest reprezentowana przez bezdrowy układ sterowania

$$\begin{cases} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{pmatrix} = \begin{bmatrix} \sin x_4 \sin x_5 & \cos x_5 \\ -\sin x_4 \cos x_5 & \sin x_5 \\ 1 & 0 \\ 0 & 1 \\ -\cos x_4 & 0 \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = G(x)u \\ y = w(x) = (x_1, x_2), \end{cases} \quad (5.4)$$

gdzie x_1 i x_2 są współrzędnymi pozycji punktu kontaktu P kuli z podłożem w układzie kartezjańskim, x_3 i x_4 są współrzędnymi sferycznymi tego punktu w układzie kuli a x_5 opisuje orientację kuli względem globalnego układu odniesienia $X_0Y_0Z_0$. W efekcie współrzędne stanu są następujące $x = (x, y, \phi, \theta, \psi)$ i reprezentują pozycję i orientację kuli w globalnym układzie odniesienia. Rysunek 5.17 przedstawia kulę kinematyczną z naniesionymi współrzędnymi i trzema układami współrzędnych: globalnym $X_0Y_0Z_0$, kuli $X_BY_BZ_B$ i układem umieszczonym w punkcie kontaktu $X_PY_PZ_P$. Należy tutaj podkreślić,



Rysunek 5.17: Kula kinematyczna

że równania (5.4) obowiązują w przypadku, gdy kula nie toczy się przez bieguny północny i południowy, zakładamy więc, że $0 < x_4 < \pi$. Obliczmy teraz macierze przybliżenia

liniowego $A(t)$ i $B(t)$ wzdłuż pary sterowanie–trajektoria $(u(t), x(t))$

$$A(t) = \frac{\partial G(x(t))u(t)}{\partial x} = \begin{bmatrix} 0 & 0 & 0 & u_1(t) \cos x_4(t) \sin x_5(t) & u_1(t) \sin x_4(t) \cos x_5(t) - u_2(t) \sin x_5(t) \\ 0 & 0 & 0 & u_1(t) \cos x_4(t) \cos x_5(t) & u_1(t) \sin x_4(t) \sin x_5(t) + u_2(t) \cos x_5(t) \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & u_1(t) \sin x_4(t) & 0 \end{bmatrix},$$

$$B(t) = G(x(t)) = \begin{bmatrix} \sin x_4(t) \sin x_5(t) & \cos x_5(t) \\ -\sin x_4(t) \cos x_5(t) & \sin x_5(t) \\ 1 & 0 \\ 0 & 1 \\ -\cos x_4(t) & 0 \end{bmatrix}.$$

Macierz wyjścia jest stała

$$C(t) = C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

Łatwo jest sprawdzić, że macierz $N(t) = CB(t)$ jest odwracalna dla każdej funkcji sterującej pod warunkiem, że $0 < x_4 < \pi$.

Będziemy rozpatrywać następujące zadanie planowania trajektorii dla kuli kinematycznej: mając zadaną trajektorię wyjścia na płaszczyźnie $y_d(t)$, znaleźć sterowanie $u(t)$, które będzie reprodukowało tę trajektorię, z określonego punktu początkowego x_0 .

Korzystając z odwrotności macierzy $N_\theta(t)$ i równania (3.13), możemy wyznaczyć równanie reprodukcji trajektorii dla kuli kinematycznej

$$\begin{aligned} & \frac{\partial^2 u(t, \theta)}{\partial t \partial \theta} + \begin{bmatrix} \alpha + 2 \operatorname{ctg} x_4(t, \theta) u_2(t, \theta) & 2 \operatorname{ctg} x_4(t, \theta) u_1(t, \theta) \\ -2 \cos x_4(t, \theta) \sin x_4(t, \theta) u_1(t, \theta) & \alpha \end{bmatrix} \frac{\partial u(t, \theta)}{\partial \theta} \\ & + \gamma \frac{\partial u(t, \theta)}{\partial t} + \begin{bmatrix} \gamma(\alpha + \operatorname{ctg} x_4(t, \theta) u_2(t, \theta)) & \gamma \operatorname{ctg} x_4(t, \theta) u_1(t, \theta) \\ -\gamma \cos x_4(t, \theta) u_1(t, \theta) & \alpha \gamma \end{bmatrix} u(t, \theta) \\ & + \begin{bmatrix} 0 & 0 & 0 & a_{14} & a_{15} \\ 0 & 0 & 0 & a_{24} & a_{25} \end{bmatrix} \xi_\theta(t) = \gamma \begin{bmatrix} \frac{\sin x_5(t, \theta)}{\sin x_4(t, \theta)} & -\frac{\cos x_5(t, \theta)}{\sin x_4(t, \theta)} \\ \cos x_5(t, \theta) & \sin x_5(t, \theta) \end{bmatrix} (\ddot{y}_d(t) + \alpha \dot{y}_d(t)), \end{aligned} \quad (5.5)$$

gdzie

$$\begin{aligned} a_{14} &= u_1(t, \theta)(\alpha \operatorname{ctg} x_4(t, \theta) - 2u_2(t, \theta)) + \operatorname{ctg} x_4(t, \theta) \dot{u}_1(t, \theta), \\ a_{15} &= \frac{1}{\sin x_4(t, \theta)} (-\cos x_4(t, \theta) \sin x_4(t, \theta) u_1^2(t, \theta) + \alpha u_2(t, \theta) + \dot{u}_2(t, \theta)), \\ a_{24} &= u_1^2(t, \theta) (-\cos^2 x_4(t, \theta) + \sin^2 x_4(t, \theta)), \\ a_{25} &= 2 \cos x_4(t, \theta) u_1(t, \theta) u_2(t, \theta) + \sin x_4(t, \theta) (\alpha u_1(t, \theta) + \dot{u}_1). \end{aligned}$$

Symbol $\dot{u}$ oznacza pochodną po czasie $\frac{\partial u(t, \theta)}{\partial t}$. Równanie (5.5) razem z równaniem kinematyki (5.4) i równaniem wariacyjnym

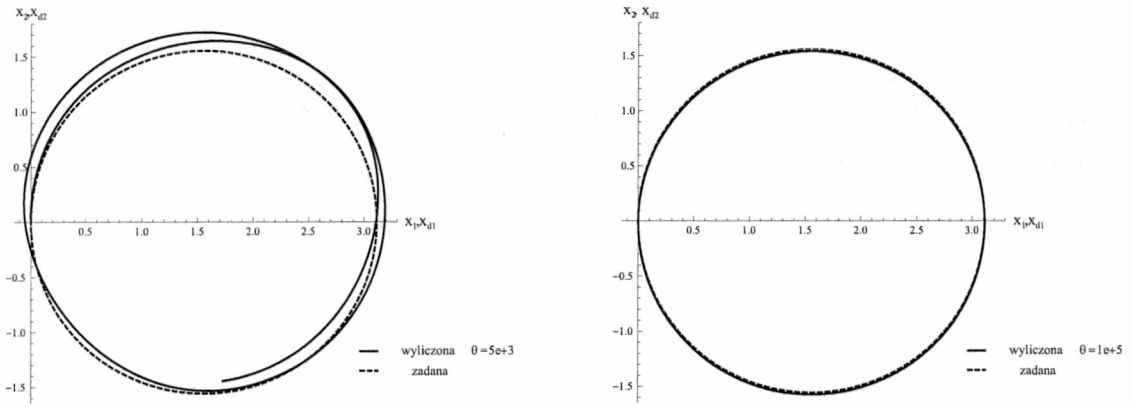
$$\begin{cases} \dot{x}_\theta(t) = G(x_\theta(t))u_\theta(t) \\ \dot{\xi}_\theta(t) = A_\theta(t)\xi_\theta(t) + B_\theta(t)\frac{\partial u(t, \theta)}{\partial \theta}, \end{cases}$$

definiuje układ reprodukcji trajektorii wyjścia dla kuli kinematycznej [21]. Układ ten powinien być rozwiązywany przy warunkach początkowych $x_\theta(0) = x_0$, $\xi_\theta(0) = 0$ oraz warunkach brzegowych $u(t, 0) = u_0(t)$ i $u(0, \theta) = u_\theta(0)$ będącego rozwiązaniem poniższego liniowego równania różniczkowego zwyczajnego

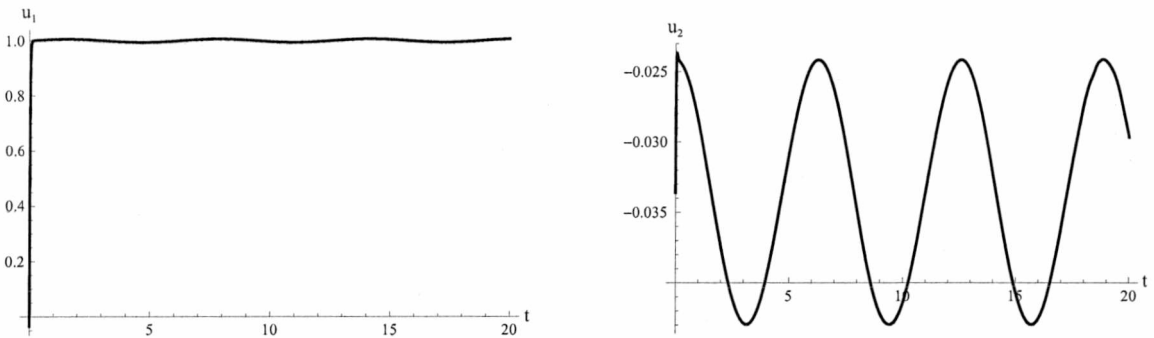
$$\frac{du(0, \theta)}{d\theta} + \gamma u(0, \theta) = \gamma \begin{bmatrix} \frac{\sin x_5(0)}{\sin x_4(0)} & -\frac{\cos x_5(0)}{\sin x_4(0)} \\ \cos x_5(0) & \sin x_5(0) \end{bmatrix} (\dot{y}_d(0) - \alpha(w(x_0)) - y_d(0)).$$

Rozwiążmy zadanie reprodukcji trajektorii. Niech zadana trajektoria będzie okręgiem $y_{d1}(t) = \tan 1 - \cos(t \cos 1) \tan 1$, $y_{d2}(t) = \sin(t \cos 1) \tan 1$; warunki początkowe $x(0) = (0, 0, 0 - 1, 0)$, czas symulacji $t_{\max} = 20$, $\gamma = 1$ i warunek początkowy na sterowanie $u_0(t) = (1, 1)$. Otrzymane rozwiązania numeryczne dla dwóch α : $\alpha = 50$ i $\alpha = 200$ oraz dwóch wartości parametru θ : $\theta = 5 \cdot 10^3$ i $\theta = 10^5$ zostały przedstawione na rysunkach 5.18–5.25. Również w tym przypadku można zaobserwować, że dla większych wartości θ otrzymane rozwiązanie jest bliższe trajektorii zadanej.

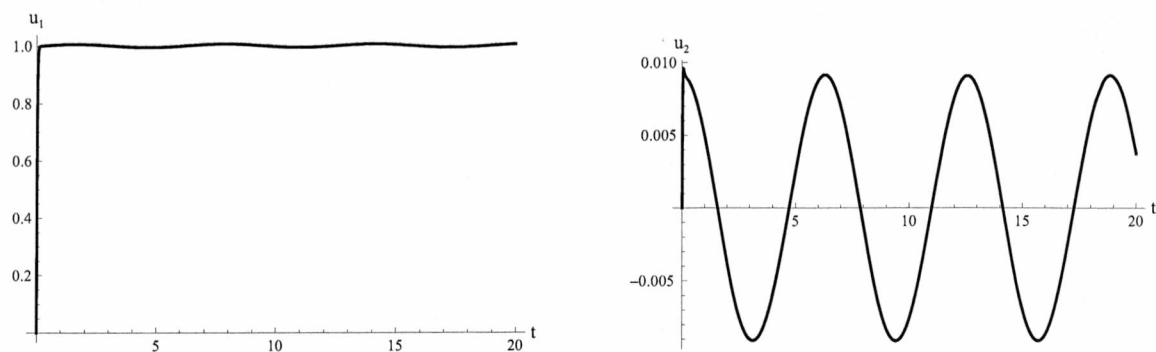
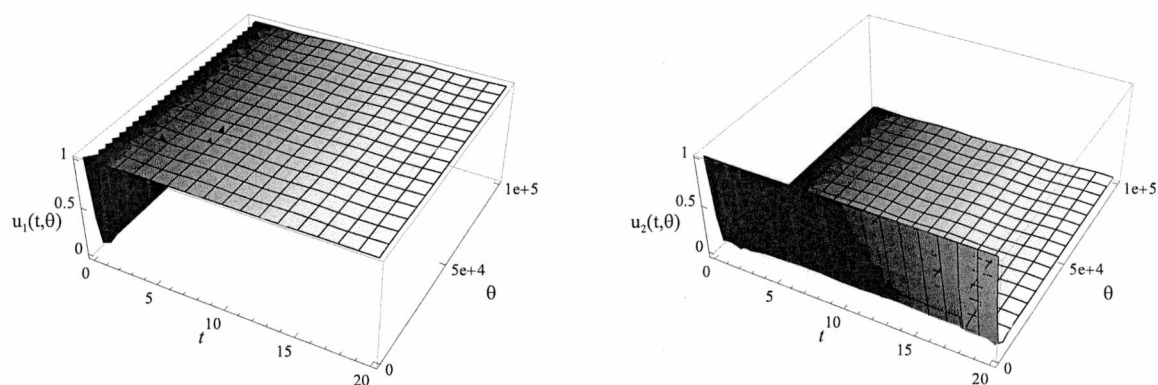
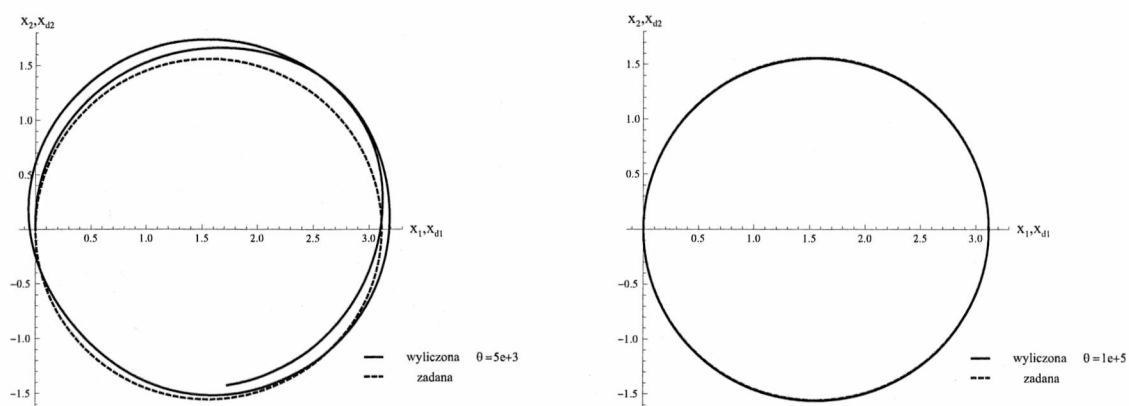
Zauważmy, że układ reprodukcji trajektorii gwarantuje reprodukcję klasy C^0 . Oznacza to, że otrzymane funkcje sterujące są bliskie rzeczywistym w sensie C^0 .

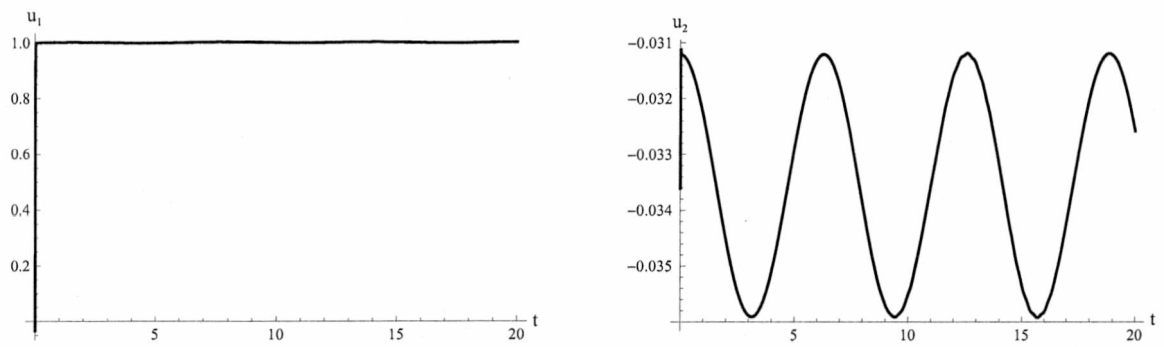
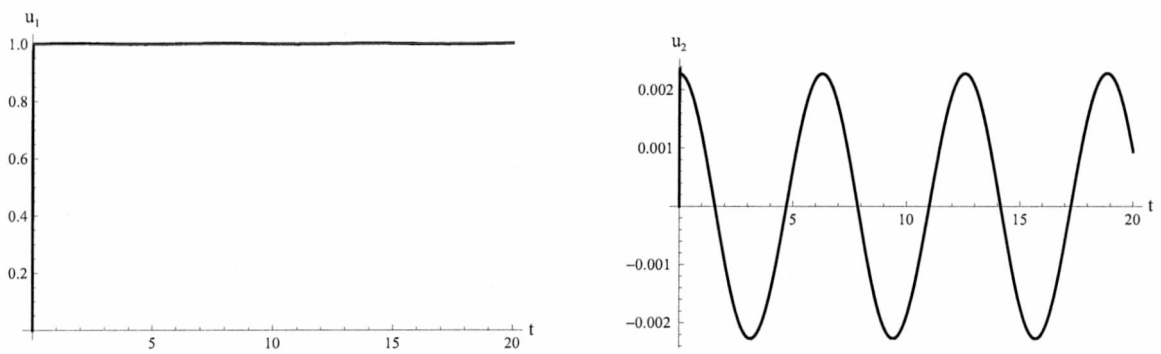
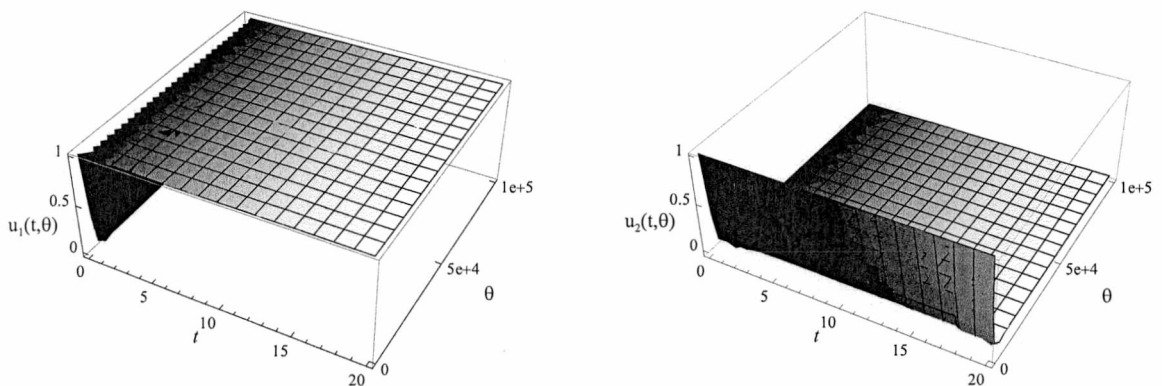


Rysunek 5.18: Trajektorie punktu kontaktu dla $\alpha = 50$, $\theta = 5 \cdot 10^3$ (z lewej) oraz dla $\alpha = 50$, $\theta = 10^5$ (z prawej)



Rysunek 5.19: Sterowania dla $\alpha = 50$, $\theta = 5 \cdot 10^3$

Rysunek 5.20: Sterowania dla $\alpha = 50$, $\theta = 10^5$ Rysunek 5.21: Rozwiązanie równania reprodukcji trajektorii dla $\alpha = 50$, $\theta = 10^5$ Rysunek 5.22: Trajektoria punktu kontaktu dla $\alpha = 200$, $\theta = 5 \cdot 10^3$ (z lewej) i dla $\alpha = 200$, $\theta = 10^5$ (z prawej)

Rysunek 5.23: Sterowania dla $\alpha = 200$, $\theta = 5 \cdot 10^3$ Rysunek 5.24: Sterowania dla $\alpha = 200$, $\theta = 10^5$ Rysunek 5.25: Rozwiązanie równania reprodukcji trajektorii dla $\alpha = 200$, $\theta = 10^5$

Rozdział 6

Podsumowanie

W pracy wyprowadzono i przebadano różne algorytmy aproksymacji algorytmów planowania ruchu robota. W pierwszej części, celem zadania aproksymacji było otrzymanie algorytmu typu jakobianu rozszerzonego, aproksymującego inny algorytm jakobianowy. Ze względu na pożądane cechy, skupiono się na aproksymacji algorytmu typu jakobianu pseudoodwrotnego. Otrzymane algorytmy posiadają, oprócz powtarzalności, szybkość zbieżność i własność minimalizacji prędkości przegubowych w trakcie ruchu, co charakteryzuje algorytm typu jakobianu pseudoodwrotnego. Wyprowadzono dwa różne kryteria aproksymacji oparte na pojęciach rachunku wariacyjnego. Udowodniono, że pierwsze z tych kryteriów może być stosowane zarówno do kinematyki wyrażonej we współrzędnych, jak i opisanej jako odwzorowanie w grupę $SE(3)$. Podstawą sformułowania zadania aproksymacji przy pomocy pojęć rachunku wariacyjnego było, w pierwszym przypadku, zdefiniowanie miary odległości pomiędzy dwoma odwrotnościami jakobianu odwrotnymi. Natomiast w drugim wypadku, funkcjonal błędu został zdefiniowany jako miara odległości odpowiednich kodystrybucji stowarzyszonych z odwrotnościami jakobianów. Przez minimalizację funkcjonalów błędu aproksymacji ze względu na funkcję rozszerzającą otrzymuje się algorytm typu jakobianu rozszerzonego będący przybliżeniem algorytmu typu jakobianu pseudoodwrotnego. Przyjęto, że we wszystkich przypadkach zadanie aproksymacji rozwiązywane jest na zbiorze nieosobliwych konfiguracji robota. Otrzymane zadania aproksymacji można rozwiązać korzystając z równania Eulera–Lagrange’a lub z metody Ritza. Równanie Eulera–Lagrange’a prowadzi do układu liniowych, eliptycznych równań różniczkowych cząstkowych. W przypadku robotów o większej liczbie stopni swobody konieczne okazało się zastosowanie metody Ritza. W celach porównawczych, przedstawiono podejście pochodzące z geometrii różniczkowej, które jest oparte na aproksymacji kodystrybucji reprezentujących odwrotności jakobianów. Celem aproksymacji było znalezienie kodystrybucji pokrywającej się z daną kodystrybucją na pewnym obszarze przestrzeni konfiguracyjnej robota.

Wszystkie uzyskane w pracy algorytmy przetestowano w badaniach symulacyjnych. Badania te pokazały, że można zbudować algorytm typu jakobianu rozszerzonego, który jest w określonym sensie bliski algorytmowi jakobianu pseudoodwrotnego. Wykazano także, że dla metody geometrycznej, jakość aproksymacji zależy od odległości trajektorii przegubowej (konfiguracyjnej) od liścia związanego z punktem początkowym. Obliczenia numeryczne pokazały, że dla pewnych zadań trajektorie przegubowe generowane przez algorytm typu jakobianu rozszerzonego otrzymanego metodą wariacyjną mogą być stosunkowo odległe od rozwiązań algorytmu typu jakobianu pseudoodwrotnego. Niemniej jednak, nawet w najgorszym przypadku, uzyskane trajektorie były bliższe niż rozwiązania otrzymywane za pomocą nieoptymalnego algorytmu typu jakobianu rozszerzonego.

Pokazano, że ponieważ aproksymację przeprowadza się nad wskazanym obszarem konfiguracji i daje ona przybliżenie w sensie średnim, zawężenie tego obszaru w sposób dedykowany konkretnemu zadaniu poprawia jakość aproksymacji. Odbywa się to jednak kosztem utraty ogólności i uniwersalności algorytmu.

W pracy zawarto wnioski dotyczące implementacji przedstawionych algorytmów. W obliczeniach korzystających z metody Ritza reprezentacja funkcji rozszerzającej w postaci wielomianu pierwszego stopnia okazała się najbardziej efektywna. Spośród dwóch reprezentacji kinematyki, mniej czasochłonne jest otrzymanie rozwiązywania zadania aproksymacji zdefiniowanego dla kinematyki w $SE(3)$. Wreszcie, porównując metodę wariacyjną i geometryczną można wywnioskować, że obie metody dają zadowalające wyniki. Zaletą zastosowania metody geometrycznej jest stosunkowo krótki czas obliczeń, ale funkcja rozszerzająca musi być wyliczana dla każdego zadania indywidualnie. Wariacyjne zadanie optymalizacji jest bardziej czasochłonne obliczeniowo, jednak otrzymana funkcja rozszerzająca jest bardziej uniwersalna.

W drugiej części rozprawy opisano zastosowanie metody homotopii do zadania reprodukcji trajektorii wyjścia lub stanu układu robotycznego. Wyprowadzono równanie reprodukcji dla afinicznego układu sterowania z wyjściem. Równanie reprodukcji trajektorii jest równaniem różniczkowym cząstkowym o zmiennej zależnej będącą funkcją sterującą. Poszukiwana funkcja sterująca, dla której układ realizuje zadaną trajektorię, jest otrzymywana w granicy przy parametrze $\theta \rightarrow +\infty$. Algorytm przetestowano w badaniach symulacyjnych. Zgodnie z oczekiwaniem, wraz ze wzrostem wartości parametru θ wzrasta dokładność reprodukcji. Jak wykazały badania, satysfakcjonujące rozwiązanie otrzymuje się często przy stosunkowo niewielkich wartościach parametru θ . Wartość tego parametru zależy od modelu, dla którego rozwiązujemy zadanie reprodukcji trajektorii. Otrzymana funkcja sterująca jest przybliżeniem rzeczywistej w sensie C^0 .

Wyniki otrzymane w pracy stanowią uzasadnienie tezy sformułowanej we Wstępie.

Na podstawie przeprowadzonych badań można nakreślić kierunki prac na przyszłość. W dziedzinie aproksymacji algorytmów jakobianowych będą to:

- rozszerzenie metody aproksymacji na inne algorytmy jakobianowe,
- uwzględnienie osobliwości algorytmicznych w funkcjonale jakości aproksymacji,
- rozszerzenie zastosowań metody na manipulatory mobilne i inne układy robotyczne,
- opracowanie metody aproksymacji wzdłuż trajektorii,
- usprawnienie obliczeń numerycznych dla metod wariacyjnych.

Jeżeli chodzi o zastosowanie metody homotopii do rozwiązania zadania reprodukcji trajektorii:

- osłabienie założeń, przy których wyprowadzono równanie reprodukcji,
- opracowanie metody reprodukcji w sensie C^k , $k \geq 1$.

Bibliografia

- [1] E. L. Allgower, K. Georg. *Numerical Continuation Methods: an Introduction*. Springer-Verlag New York, Inc., Nowy Jork, 1990.
- [2] E. L. Allgower, K. Georg. *Introduction to Numerical Continuation Methods*. Society for Industrial and Applied Mathematics, 2003.
- [3] F. Alouges, Y. Chitour, R. Long. A motion planning algorithm for the rolling-body problem. *Proc. 48th IEEE CDC/CCC Conference*, strony 2112–2116, grudzień 2009.
- [4] J. Angeles. *Fundamentals of Robotic Mechanical Systems*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, wydanie 2, 2002.
- [5] V. Arnol'd, M. Levi. *Geometrical Methods in the Theory of Ordinary Differential Equations*. Grundlehren der mathematischen Wissenschaften. Springer-Verlag, 1988.
- [6] S. Axler, P. Bourdon, W. Ramey. *Harmonic Function Theory*. Springer, 2001.
- [7] J. Baillieul. Kinematic programming alternatives for redundant manipulators. *Proc. IEEE Int. Conf. Rob. Automat.*, strony 818–823, St. Louis, 1985.
- [8] A. Bloch. *Nonholonomic Mechanics and Control*. Interdisciplinary applied mathematics: Systems and control. Springer-Verlag, 2003.
- [9] R. Brockett, M. Mesarovic. The reproducibility of multivariable systems. *J. Maths. Analysis and Applications*, 11:548–563, 1965.
- [10] P. Chang. A closed-form solution for inverse kinematics of robot manipulators with redundancy. *IEEE Journal on Robotics and Automation*, 3(5):393–403, 1987.
- [11] A. Chelouah, Y. Chitour. On the motion planning of rolling surfaces. *Forum Mathematicum*, 15(5):727–758, 2006.
- [12] S. Chiaverini, G. Oriolo, I. D. Walker. Kinematically redundant manipulators. *Springer Handbook of Robotics*, strony 245–268. Springer-Verlag, Berlin, 2008.
- [13] Y. Chitour. A continuation method for motion-planning problems. *ESAIM: Control, Optimisation and Calculus of Variations*, 12:139–168, 2006.
- [14] Y. Chitour, H. J. Sussmann. Line-integral estimates and motion planning using the continuation method. J. Baillieul, S. S. Sastry, H. J. Sussmann, redaktorzy, *Essays on Mathematical Robotics*, strony 91–125. Springer-Verlag, 1998.

- [15] A. W. Divelbiss, S. Seereeram, J. T. Wen. Kinematic path planning for robots with holonomic and nonholonomic constraints. J. Baillieul, S. Sastry, H. Sussmann, redaktorzy, *Essays on Mathematical Robotics*, strony 127–150, New York, 1998. Springer-Verlag.
- [16] A. W. Divelbiss, J. T. Wen. A path space approach to nonholonomic motion planning in the presence of obstacles. *IEEE Trans. Robot. and Autom.*, 13(3):443–451, 1997.
- [17] I. Gel'fand, S. Fomin, R. Silverman. *Calculus of Variations*. Selected Russian publications in the mathematical sciences. Prentice-Hall, 1965.
- [18] P. Hartman. *Ordinary differential equations*. Birkhäuser, Boston, 1973.
- [19] S. Hildebrandt, M. Giaquinta. *Calculus of Variations I*. Springer, 2004.
- [20] C. Hillermeier. *Nonlinear Multiobjective Optimization: a Generalized Homotopy Approach*. International series of numerical mathematics. Birkhäuser Verlag, 2001.
- [21] J. Karpińska, K. Tchoń. Continuation method approach to trajectory planning in robotic systems. *Proc. 16th International Conference on Methods and Models in Automation and Robotics*, strony 51–56, 2011.
- [22] J. Karpińska, K. Tchoń. Optimal extended Jacobian inverse kinematics algorithm with application to attitude control of robotic manipulators. K. Kozłowski, redaktor, *Robot motion and control 2011; Lecture Notes in Control and Information Sciences*. Springer-Verlag, 2011. w druku.
- [23] J.-O. Kim, P. Khosla. Real-time obstacle avoidance using harmonic potential functions. *IEEE Transactions on Robotics and Automation*, czerwiec 1992.
- [24] C. A. Klein, L.-C. Chu. Comparison of extended Jacobian and Lagrange multiplier based methods for resolving kinematic redundancy. *Journal of Intelligent and Robotic Systems*, 19(1):39–54, 1997.
- [25] C. A. Klein, C. Chu-Jeng, S. Ahmed. A new formulation of the extended Jacobian method and its use in mapping algorithmic singularities for kinematically redundant manipulators. *IEEE Trans. Robot. Automat.*, 11:50–55, 1995.
- [26] C. A. Klein, C. H. Huang. Review of pseudoinverse control for use with kinematically redundant manipulators. *IEEE Transactions on Systems Man and Cybernetics*, SMC-13(3):245–250, 1983.
- [27] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006.
- [28] R. Murray, Z. Li, S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.
- [29] R. M. Murray, S. S. Sastry, L. Zexiang. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Inc., Boca Raton, FL, USA, 1994.
- [30] K. Nazarczuk, C. Zieliński. POLYCRANK – fast robot without joint limits. *Proc. ROMANSY 12*, strony 317–324, Wiedeń, 1998. Springer-Verlag.

- [31] K. Nazarczuk, C. Zieliński, S. Łuszczak. Development of the design of POLYCRANK manipulator without joint limits. *Proc. ROMANSY 13*, strony 285–292, Wiedeń, 2000. Springer-Verlag.
- [32] F. C. Park, R. W. Brockett. Kinematic dexterity of robotic mechanisms. *I. J. Robotic Res.*, 13(1):1–15, 1994.
- [33] H. Ramirez, S. Agrawal. *Differentially Flat Systems*. Marcel Dekker, 2004.
- [34] A. Ratajczak, J. Karpińska, K. Tchoń. Algorytm planowania ruchu statku z priorytetowaniem zadań: metoda endogenicznej przestrzeni konfiguracyjnej. *Prace Naukowe - Politechnika Warszawska. Elektronika*, wolumen 166, strony 535–544, X KKR, Piechowice, Polska, 2008.
- [35] A. Ratajczak, J. Karpińska, K. Tchoń. Task-priority motion planning of underactuated systems: an endogenous configuration space approach. *Robotica*, 28(6):885–892, 2010.
- [36] W. Respondek. Right and left invertibility of nonlinear control systems. H. Susmann, redaktor, *Nonlinear Controllability and Optimal Control*, strony 133–176. Marcel Dekker, Nowy Jork, 1990.
- [37] S. L. Richter, R. A. DeCarlo. Continuation methods: Theory and applications. *IEEE Trans. Circuits and Syst.*, 30:337–352, 1983.
- [38] R. Roberts, A. A. Maciejewski. Repeatable generalized inverse control strategies for kinematically redundant manipulators. *IEEE Trans. Automat. Contr*, 38:689–699, 1993.
- [39] R. G. Roberts, A. A. Maciejewski. Nearest optimal repeatable control strategies for kinematically redundant manipulators. *IEEE Trans. Robotics Automat*, 8:327–337, 1992.
- [40] R. G. Roberts, A. A. Maciejewski. Calculation of repeatable control strategies for kinematically redundant manipulators. *Journal of Intelligent and Robotic Systems*, 14(1):105–130, 1995.
- [41] S. Seereeram, J. Wen. A global approach to path planning for redundant manipulators. *IEEE Trans. Robot. and Autom.*, 11(1):152–160, 1995.
- [42] J. Selig. *Geometric Fundamentals of Robotics*. Springer-Verlag, 2005.
- [43] T. Shamir, Y. Yomdin. Repeatability of redundant manipulators: Mathematical solution of the problem. *IEEE Transactions on Automatic Control*, 33:1004–1009, 1988.
- [44] L. Silverman. Inversion of multivariable linear systems. *Automatic Control, IEEE Transactions on*, 14(3):270 – 276, czerwiec 1969.
- [45] S. Singh. Functional reproducibility of multivariable nonlinear systems. *Automatic Control, IEEE Transactions on*, 27(1):270 – 272, luty 1982.

- [46] W. M. Sluis, A. Banaszuk, J. Hauser, R. M. Murray. A homotopy algorithm for approximating geometric distributions by integrable systems. *Systems & Control Lett.*, 27:285–291, 1995.
- [47] E. D. Sontag. *Mathematical Control Theory*. Springer-Verlag, Nowy Jork, 1990.
- [48] M. Spong, M. Vidyasagar, Z. Pilat. *Dynamika i sterowanie robotów*. Wydawnictwa Naukowo-Techniczne, 1997.
- [49] R. S. Strichartz. Sub-Riemannian geometry. *J. Differential Geom.*, 24(2):221–263, 1986.
- [50] H. J. Sussmann. New differential geometric methods in nonholonomic path finding. *Progress Systems and Control Theory*, strony 365–384, 1992.
- [51] H. J. Sussmann. A continuation method for nonholonomic path-finding problems. *Proc. IEEE CDC/ICC Conference*, strony 2717–2723, 1993.
- [52] K. Tchoń. Continuation method in robotics. *Proc. 7th Conference on Computer Methods and Systems*, strony 17–24, Kraków, Polska, 2007.
- [53] K. Tchoń. Optimal extended Jacobian inverse kinematics algorithms for robotic manipulators. *IEEE Transactions on Robotics*, 24:1440–1445, 2008.
- [54] K. Tchoń, M. Janiak. Repeatable approximation of the Jacobian pseudo-inverse. *Systems & Control Letters*, 58(12):849–856, 2009.
- [55] K. Tchoń, J. Karpińska. Asymptotyczna reprodukcja trajektorii stanu robota. *Prace Naukowe - Politechnika Warszawska. Elektronika*, 175:457–466, 2010.
- [56] K. Tchoń, J. Karpińska, M. Janiak. Approximation of Jacobian inverse kinematics algorithms. *Int. J. Appl. Math. Comput. Sci.*, 19(4):519–531, 2009.
- [57] K. Tchoń, J. Jakubiak. Endogenous configuration space approach to mobile manipulators: A derivation and performance assessment of Jacobian inverse kinematics algorithms. *International Journal of Control*, 76(14):1387–1419, 2003.
- [58] K. Tchoń, J. Jakubiak. Extended Jacobian inverse kinematics algorithm for nonholonomic mobile robots. *International Journal of Control*, 79(8):895–909, 2006.
- [59] K. Tchoń, A. Mazur, I. Duleba, R. Hossa, R. Muszyński. *Manipulatory i roboty mobilne*. Problemy współczesnej nauki teoria i zastosowania: Robotyka. Akademicka Oficyna Wydawnicza PLJ, Warszawa, Polska, 2000.
- [60] L. T. Watson. Theory of globally convergent probability-one homotopies for nonlinear programming. *SIAM J. on Optimization*, 11:761–780, 2000.
- [61] L. T. Watson, R. T. Haftka. Modern homotopy methods in optimization. *Comput. Methods Appl. Mech. Eng.*, 74:289–305, 1989.