

PROBLEMY WSPÓŁCZESNEJ NAUKI  
TEORIA I ZASTOSOWANIA

ROBOTYKA

Krzysztof Tchoń

Alicja Mazur

Robert Hossa

Ignacy Dulęba

Robert Muszyński

# MANIPULATORY I ROBOTY MOBILNE

Modele  
planowanie ruchu  
sterowanie



Akademicka Oficyna Wydawnicza PLJ  
Warszawa 2000

Książka zawiera monograficzny wykład podstaw robotyki obejmujący kinematykę, dynamikę, planowanie ruchu i sterowanie manipulatorów i robotów mobilnych. Szczególne miejsce zajmują w książce zagadnienia modelowania kinematyki i dynamiki robotów, osobliwości kinematyki, algorytmy regularnej i osobliwej kinematyki odwrotnej, algorytmy sterowania manipulatorów sztywnych i elastycznych oraz robotów mobilnych przy ograniczonej znajomości modelu dynamiki, a także algorytmy planowania ruchu robotów mobilnych wykorzystujące metody geometrycznej teorii sterowania. W zakresie tych zagadnień wyniki własne autorów zostały przedstawione w kontekście wyników udokumentowanych w literaturze przedmiotu.

Książka stanowi świadectwo dorobku szkoły uprawiania robotyki, jaka powstała w Zakładzie Podstaw Cybernetyki i Robotyki Instytutu Cybernetyki Technicznej Politechniki Wrocławskiej. Do jej adresatów należą studenci kierunku Automatyka i Robotyka politechnik, konstruktorzy robotów i projektanci systemów automatyki i robotyki, inżynierowie zajmujący się automatyzacją i robotyzacją, a także pracownicy naukowcy i doktoranci z dyscyplin Automatyka i Robotyka, Informatyka, Matematyka Stosowana.

# Manipulatory i roboty mobilne

Modele, planowanie ruchu, sterowanie

Książka zostaje udostępniona w wersji pdf za zgodą  
Oficyny Wydawniczej PLJ i może być kopiowana wyłącznie  
w całości, razem ze stroną tytułową i tym przypisem.

Kompilacja: 23 września 2021

**PROBLEMY WSPÓŁCZESNEJ NAUKI  
TEORIA I ZASTOSOWANIA**

**ROBOTYKA**

---

Edytor serii: Leonard Bolc

Krzysztof Tchoń  
Alicja Mazur Ignacy Dulęba  
Robert Hossa Robert Muszyński

# **Manipulatory i roboty mobilne**

Modele, planowanie ruchu, sterowanie

Akademicka Oficyna Wydawnicza PLJ  
Warszawa 2000

© Copyright by Ignacy Dulęba, Robert Hossa, Alicja Mazur,  
Robert Muszyński, Krzysztof Tchoń  
Wrocław 2000

© Copyright by Akademicka Oficyna Wydawnicza PLJ  
Warszawa 2000

### **Autorzy**

*Ignacy Dulęba, Robert Hossa, Alicja Mazur,  
Robert Muszyński, Krzysztof Tchoń,  
Zakład Podstaw Cybernetyki i Robotyki,  
Instytut Cybernetyki Technicznej, Politechnika Wrocławska*

### **Recenzent**

*Prof. dr hab. inż. Krzysztof Kozłowski, Politechnika Poznańska*

### **Redaktor**

*Anna Bittner*

### **Komputerowy skład tekstu**

*Robert Muszyński*

### **Dobór koloru okładki**

*Ula Tchoń*

### **Projekt graficzny serii**

*Akademicka Oficyna Wydawnicza PLJ*

**ISBN 83-7101-427-9**

# Spis rzeczy

Spis oznaczeń	9
<b>1 Wprowadzenie</b>	<b>15</b>
Literatura	20
<b>I Kinematyka manipulatorów i robotów mobilnych</b>	<b>23</b>
<b>2 Modele kinematyki układów robotycznych</b>	<b>25</b>
2.1 Ruch ciała sztywnego	25
2.2 Kinematyka układu robotycznego	36
2.3 Kinematyka manipulatora	40
2.3.1 Reprezentacja Denavita-Hartenberga	41
2.3.2 Reprezentacja wykładnicza	51
2.3.3 Kinematyka we współrzędnych	55
2.3.4 Jakobiany	61
2.3.5 Konfiguracje osobliwe	79
2.4 Kinematyka robota mobilnego	83
2.5 Komentarze i uwagi bibliograficzne	88
Literatura	90
<b>3 Algorytmy kinematyki odwrotnej manipulatora</b>	<b>95</b>
3.1 Regularne odwrotne zadanie kinematyki	97
3.1.1 Bezpośrednie podejście algebraiczne	98
3.1.2 Podejście geometryczne	101
3.1.3 Metody jakobianowe	103
3.1.4 Metoda mnożników Lagrange'a	125
3.1.5 Elipsoida manipulowalności	128
3.2 Osobliwe odwrotne zadanie kinematyki	131
3.2.1 Unikanie osobliwości	131

3.2.2	Metoda postaci normalnych . . . . .	140
3.2.3	Metoda jakobianu dołączonego . . . . .	163
3.2.4	Metoda przestrzeni zerowej . . . . .	166
3.2.5	Metoda jakobianu odpornego . . . . .	168
3.3	Komentarze i uwagi bibliograficzne . . . . .	174
	Literatura . . . . .	176
<b>4</b>	<b>Kinematyka odwrotna robotów mobilnych</b>	<b>181</b>
4.1	Metoda Newtona . . . . .	181
4.2	Elipsoida mobilności . . . . .	186
4.3	Komentarze i uwagi bibliograficzne . . . . .	186
	Literatura . . . . .	188
<b>II</b>	<b>Algorytmy sterowania manipulatorów</b>	<b>191</b>
<b>5</b>	<b>Algorytmy sterowania manipulatorów sztywnych w przestrzeni zadaniowej</b>	<b>193</b>
5.1	Dynamika układu robotycznego . . . . .	194
5.2	Dynamika manipulatora sztywnego . . . . .	196
5.3	Sterowanie w przestrzeni zadaniowej manipulatora . . . . .	202
5.3.1	Linearyzacja i odsprzęganie wejściowo-wyjściowe . . . . .	203
5.3.2	Transformacja do zadania sterowania w przestrzeni przegubowej . . . . .	206
5.4	Komentarze i uwagi bibliograficzne . . . . .	206
	Literatura . . . . .	207
<b>6</b>	<b>Algorytmy sterowania manipulatorów sztywnych w przestrzeni przegubowej</b>	<b>209</b>
6.1	Algorytmy wymagające pełnej znajomości modelu . . . . .	213
6.1.1	Algorytmy typu obliczanego momentu . . . . .	213
6.1.2	Algorytmy typu dysypatywnego . . . . .	216
6.2	Algorytmy sterowania przy parametrycznej nieznajomości modelu . . . . .	222
6.2.1	Algorytmy adaptacyjne typu obliczanego momentu . . . . .	224
6.2.2	Algorytmy adaptacyjne typu dysypatywnego . . . . .	228
6.3	Algorytmy sterowania przy strukturalnej nieznajomości modelu . . . . .	231
6.3.1	Algorytm sterowania ślizgowego . . . . .	232



6.3.2	Algorytm Qu-Dorseya — regulator PD o stałym wzmo- cnieniu . . . . .	235
6.3.3	Algorytm lambda-śledzenia — regulator PD o dyna- micznym wzmocnieniu . . . . .	236
6.4	Komentarze i uwagi bibliograficzne . . . . .	239
	Literatura . . . . .	241
<b>7</b>	<b>Algorytmy sterowania manipulatorów o elastycznych prze- gubach</b> . . . . .	<b>245</b>
7.1	Dynamika manipulatora o elastycznych przegubach . . . . .	246
7.2	Algorytmy sterowania . . . . .	251
7.2.1	Algorytm linearyzacji statycznej . . . . .	251
7.2.2	Algorytm całkowania wstecznego . . . . .	254
7.2.3	Algorytm Ortegi-Lorii . . . . .	259
7.3	Komentarze i uwagi bibliograficzne . . . . .	261
	Literatura . . . . .	262
<b>III</b>	<b>Algorytmy planowania ruchu i sterowania robotów mobilnych</b> . . . . .	<b>263</b>
<b>8</b>	<b>Zadanie planowania ruchu</b> . . . . .	<b>265</b>
8.1	Pojęcia podstawowe . . . . .	265
8.2	Technika nawiasu Liego . . . . .	273
8.3	Komentarze i uwagi bibliograficzne . . . . .	280
	Literatura . . . . .	281
<b>9</b>	<b>Metody planowania ruchu ogólnego przeznaczenia</b> . . . . .	<b>285</b>
9.1	Metoda oparta na Zasadzie Maksimum Pontriagina . . . . .	286
9.2	Metoda Newtona . . . . .	287
9.3	Metoda uśredniania . . . . .	292
9.4	Metoda Lie-algebraiczna . . . . .	297
9.4.1	Uogólniona formuła Campbella-Bakera-Hausdorffa- -Dynkina . . . . .	299
9.4.2	Algorytm metody Lie-algebraicznej . . . . .	305
9.4.3	Znaczenie uogólnionej formuły Campbella-Bakera- -Hausdorffa-Dynkina . . . . .	307
9.5	Komentarze i uwagi bibliograficzne . . . . .	309
	Literatura . . . . .	312

<b>10 Specjalizowane metody planowania ruchu</b>	<b>317</b>
10.1 Optymalne planowanie ruchu jednokołowego robota mobilnego . . . . .	317
10.1.1 Rodziny ekstremal dla zadania Reedsa-Sheppa . . . . .	318
10.1.2 Synteza trajektorii optymalnej . . . . .	325
10.1.3 Planowanie ruchu jednokołowego robota mobilnego poruszającego się do przodu . . . . .	329
10.1.4 Podsumowanie . . . . .	330
10.2 Metoda sterowań sinusoidalnych . . . . .	331
10.3 Metoda bazująca na twierdzeniu Stokesa . . . . .	336
10.4 Metoda osiągania podcelów . . . . .	341
10.5 Komentarze i uwagi bibliograficzne . . . . .	347
Literatura . . . . .	349
<b>11 Modele i algorytmy sterowania kołowych robotów mobilnych</b>	<b>351</b>
11.1 Dynamika układu robotycznego z ograniczeniami . . . . .	352
11.2 Modele kołowych robotów mobilnych . . . . .	353
11.2.1 Ograniczenia fazowe . . . . .	354
11.2.2 Modele kinematyki prostych kołowych robotów mobilnych . . . . .	358
11.2.3 Modele kinematyki złożonych kołowych robotów mobilnych . . . . .	366
11.2.4 Modele dynamiki . . . . .	367
11.3 Algorytmy sterowania kołowych robotów mobilnych . . . . .	369
11.3.1 Algorytm Corona-Pometa . . . . .	370
11.3.2 Algorytm linearyzacji dynamicznej . . . . .	373
11.3.3 Algorytm Walsh-Tilbury'ego-Sastry'ego-Murray-Laumonda . . . . .	377
11.3.4 Sterowanie we współrzędnych linearyzujących . . . . .	379
11.3.5 Sterowanie adaptacyjne we współrzędnych linearyzujących . . . . .	386
11.3.6 Uniwersalny adaptacyjny $\lambda$ -śledzący algorytm sterowania . . . . .	389
11.4 Komentarze i uwagi bibliograficzne . . . . .	392
Literatura . . . . .	394

---

<b>Dodatki</b>	<b>399</b>
<b>A Podstawowe pojęcia matematyczne</b>	<b>401</b>
A.1 Algebra liniowa . . . . .	401
A.2 Funkcje i odwzorowania . . . . .	403
A.3 Układy dynamiczne, pola wektorowe, nawiasy Liego . . . . .	404
A.4 Układy sterowania . . . . .	406
A.5 Zasada Maksimum Pontriagina . . . . .	408
A.6 Równania Eulera-Lagrange'a . . . . .	409
Literatura . . . . .	409
<b>B Wybrane twierdzenia o stabilności</b>	<b>411</b>
B.1 Układy autonomiczne . . . . .	411
B.2 Układy nieautonomiczne . . . . .	412
B.2.1 II Metoda Lapunowa . . . . .	412
B.2.2 Lemat Barbalata . . . . .	413
B.2.3 Twierdzenie La Salle'a-Yoshizawy . . . . .	413
B.2.4 Lemat Wena-Bayarda . . . . .	413
Literatura . . . . .	414
<b>C Wybrane dowody stabilności algorytmów sterowania</b>	<b>417</b>
C.1 Algorytm Sadegha-Horowitza . . . . .	417
C.2 Adaptacyjny algorytm Slotine'a-Li . . . . .	419
Literatura . . . . .	421
<b>Indeks</b>	<b>423</b>



# Spis oznaczeń

Oznaczenia zostały podporządkowane następującej zasadzie ogólnej: symbole niewytłuszczone (typu  $x$ ) odnoszą się do wielkości skalarnych, wytłuszczone (jak  $\mathbf{x}$ ,  $\mathbf{X}$ ,  $\mathbf{X}$ ) oznaczają wektory, macierze lub zbiory. Symbole postaci  $\mathbb{X}$  oznaczają wyróżnione klasy obiektów matematycznych, a symbole typu  $\mathcal{X}$  odnoszą się do pojęć o znaczeniu specjalnym w kontekście robotyki. Liczby w nawiasach podają numer strony, na której dane oznaczenie zostało wprowadzone lub użyte po raz pierwszy.

$\simeq$	— równość przybliżona (298)
$\cong$	— równoważność (29)
$\equiv$	— równość tożsamościowa (146)
$\  \ \ $	— norma euklidesowa wektora (25)
$\  \ \ $	— norma macierzy (402)
$[ \ ]$	— macierz skośnie symetryczna (31)
$[\mathbf{X}, \mathbf{Y}]$	— nawias Liego (38)
$\mathbf{A}^T$	— transpozycja macierzy (28)
$\mathbf{A}^{-1}$	— odwrotność macierzy (29)
$\mathbf{A}^\#$	— pseudoodwrotność macierzy (402)
$\mathbf{A} > 0$	— macierz dodatnio określona (205)
$\mathbf{f} _{\mathbf{u}}$	— obcięcie odwzorowania w dziedzinie (146)
$\mathbf{x}^T$	— transpozycja wektora (25)
$\mathbf{x} \cdot \mathbf{y}$	— iloczyn skalarny (25)
$\mathbf{x}^T \mathbf{y}$	— iloczyn skalarny (25)
$\langle \mathbf{x}, \mathbf{y} \rangle$	— iloczyn skalarny (25)
$\mathbf{x} \times \mathbf{y}$	— iloczyn wektorowy (25)
$\mathbf{U} \times \mathbf{V}$	— iloczyn kartezjański zbiorów (41)
$\varphi \circ \psi$	— złożenie odwzorowań (55)
$\mathbf{A}$	— macierz ograniczeń fazowych w postaci Pfaffa (36)
$\mathbf{A}_i^j$	— transformacja układu $i$ w układ $j$ (43)
$\text{ad}_X^i \mathbf{Y}$	— iterowany nawias Liego (406)
$\text{adj} \mathbf{A}$	— macierz dołączona (401)
$\text{atan2}$	— funkcja atan2 (99)
$\mathbb{B}$	— klasa funkcji ograniczonych (403)
$\overline{\mathbf{B}}_\lambda(\mathbf{0})$	— kula domknięta (236)

- b** — wektor binormalny (26)
- C** — macierz sił Coriolisa manipulatora (195)
- $\mathbb{C}^k$  — klasa funkcji gładkich rzędu  $k$  (403)
- $\mathbb{C}^\infty$  — klasa funkcji gładkich (403)
- $\mathbb{C}^\omega$  — klasa funkcji analitycznych (403)
- $c_{kj}^i$  — symbole Christoffela I rodzaju (195)
- codim — kowymiar (80)
- corank — korząd (79)
- D** — różniczka Gâteaux (84)
- D** — wektor sił grawitacji manipulatora (195)
- $D\mathbf{k}_{\mathbf{q}_0, \Gamma}$  — jacobian analityczny robota mobilnego (86)
- $(D\mathbf{k}_{\mathbf{q}_0, \Gamma})^\#$  — pseudoodwrotność jacobianu analitycznego robota mobilnego (183)
- $d$  — różniczka (406)
- det — wyznacznik macierzy (25)
- diag — macierz diagonalna (214)
- dim — wymiar (30)
- div — dywergencja (136)
- $\Delta$  — wyróżnik konfiguracji osobliwej (134)
- $\delta_m$  — stopień mobilności robota mobilnego (361)
- $\delta_s$  — stopień sterowalności robota mobilnego (361)
- $E_m$  — elipsoida manipulowalności (128)
- $E_m^{\mathbf{q}_0, \Gamma}$  — elipsoida mobilności (186)
- e** — błąd śledzenia (205)
- $\mathbf{e}_i$  —  $i$ -ty wektor bazowy przestrzeni liniowej (401)
- exp — odwzorowanie wykładnicze (35)
- exp — strumień pola wektorowego (278)
- $\Phi$**  — macierz fundamentalna liniowego układu dynamicznego (86)
- $\varphi_t$  — strumień układu dynamicznego (404)
- $\mathcal{G}$  — dystrybucja (37)
- G** — pola stowarzyszone z ograniczeniami fazowymi w postaci Pfaffa (37)
- $\Gamma$  — indeks konfiguracji osobliwej (138)
- $\mathcal{H}$  — baza Ph. Halla (277)
- H** — hamiltonian (406)
- $\mathcal{I}$  — przedział czasu (95)
- I** — macierz momentów bezwładności silników (249)
- $I_i$  — moment bezwładności wirnika  $i$ -tego silnika (199)
- $\mathbb{I}_n$  — macierz jednostkowa rozmiaru  $n \times n$  (29)

- $\text{id}_{\mathbb{R}^n}$  — odwzorowanie identycznościowe na  $\mathbb{R}^n$  (403)  
 $\text{inf}$  — infimum (212)  
 $\mathcal{J}$  — wskaźnik jakości sterowania (408)  
 $J^a$  — jacobian analityczny manipulatora (61)  
 $J^{ae}$  — rozszerzony jacobian analityczny manipulatora (120)  
 $J^{a\lambda}$  — odwrotność przybliżona jacobianu analitycznego manipulatora (168)  
 $J_W^{a\#}$  — pseudoodwrotność jacobianu analitycznego manipulatora (112)  
 $J^b$  — jacobian geometryczny w ciele (64)  
 $J^m$  — jacobian manipulatora (68)  
 $J^s$  — jacobian geometryczny w przestrzeni (64)  
 $J^{se}$  — rozszerzony jacobian geometryczny w przestrzeni (125)  
 $J_W^{s\#}$  — pseudoodwrotność jacobianu geometrycznego w przestrzeni (125)  
 $J_{L_i}$  — macierz inercji  $i$ -tego ramienia manipulatora (197)  
 $J_{M_i}$  — macierz inercji wirnika  $i$ -tego silnika (247)  
 $\mathbb{K}$  — klasa funkcji dodatnich, ściśle rosnących (412)  
 $K$  — energia kinetyczna (194)  
 $\mathbf{K}$  — kinematyka manipulatora (39)  
 $\mathbf{K}|_{\mathbf{u}}$  — obcięcie kinematyki w dziedzinie (55)  
 $\mathbf{k}$  — kinematyka manipulatora we współrzędnych (55)  
 $\mathbf{k}_0$  — postać normalna kinematyki (141)  
 $\mathbf{k}_{q_0, T}$  — kinematyka robota mobilnego (84)  
 $\text{Ker}$  — przestrzeń zerowa (37)  
 $\kappa$  — stopień uwarunkowania (130)  
 $L$  — lagranżian (194)  
 $\mathbb{L}_n^2[0, T]$  — przestrzeń funkcji całkowalnych z kwadratem (403)  
 $\mathbb{L}_k$  — klasa funkcji całkowalnych z  $k$ -tą potęgą (403)  
 $\mathbb{L}_\infty$  — klasa funkcji ograniczonych prawie wszędzie (403)  
 $\bar{\lambda}_A$  — największa wartość własna macierzy (402)  
 $\underline{\lambda}_A$  — najmniejsza wartość własna macierzy (402)  
 $\mathcal{M}$  — rozmaitość gładka (30)  
 $\mathbf{M}$  — macierz manipulowalności (128)  
 $\mathbf{M}_{q_0, T}$  — macierz mobilności (186)  
 $\mathbf{m}$  — manipulowalność (129)  
 $\mathbf{m}_{q_0, T}$  — mobilność (186)  
 $\text{max}$  — maksimum (286)  
 $\text{mod}$  — modulo (149)  
 $N$  — funkcja Nussbauma (212)  
 $\mathbf{n}$  — wektor normalny (26)

- $o^n$  — mała funkcja rzędu  $n$  (153)  
 $\Omega^b$  — prędkość kątowa w układzie ciała (31)  
 $\Omega^s$  — prędkość kątowa w układzie przestrzeni (31)  
 $\omega_b$  — wektor prędkości kątowej w układzie ciała (32)  
 $\omega_s$  — wektor prędkości kątowej w układzie przestrzeni (32)
- P** — macierz struktury kinematycznej robota mobilnego (361)  
 **$\psi$**  — zmienna dołączona (408)
- $\mathcal{Q}$  — rozmaitość konfiguracyjna (przegubowa) (36)  
**Q** — macierz inercji manipulatora (195)
- R** — macierz obrotu (28)  
 $\mathbb{R}$  — zbiór liczb rzeczywistych (25)  
 $\mathbb{R}_+$  — zbiór liczb rzeczywistych nieujemnych (378)  
 $\mathbb{R}^n$  — przestrzeń euklidesowa  $n$ -wymiarowa (25)  
 rank — rząd (30)  
**Rot** — elementarny obrót (43)
- S** — zbiór konfiguracji osobliwych (79)  
 $S^1$  — okrąg jednostkowy (41)  
**s** — zmienna ślizgu (216)  
 $SE(2)$  — specjalna grupa euklidesowa w  $\mathbb{R}^2$  (39)  
 $SE(3)$  — specjalna grupa euklidesowa w  $\mathbb{R}^3$  (29)  
 $se(3)$  — algebra Liego specjalnej grupy euklidesowej w  $\mathbb{R}^3$  (32)  
 sgn — signum (232)  
 $SO(3)$  — specjalna grupa obrotów (28)  
 span — obiekt rozpięty (37)  
 sup — supremum (212)  
 $\sigma$  — permutacja (299)
- T** — wektor przesunięcia (28)  
 **$\mathcal{TM}$**  — wiązka styczna (30)  
 $\mathbb{T}^r$  — torus  $r$ -wymiarowy (41)  
 $\mathbb{T}_r(t)$  — sympleks  $r$ -wymiarowy (299)  
**t** — wektor styczny (26)  
 tr — ślad (34)  
**Trans** — elementarne przesunięcie (43)  
 $\theta$  — wektor parametrów modelu dynamiki (223)  
 $\hat{\theta}$  — wektor estymat parametrów modelu dynamiki (224)
- U** — zbiór otwarty (30)  
 $u(\cdot)$  — sterowanie (84)



- 
- $V$  — energia potencjalna (194)
- $V$  — funkcja Lapunowa (214)
- $V$  — zbiór otwarty (30)
- $V^b$  — prędkość w układzie ciała (31)
- $V^s$  — prędkość w układzie przestrzeni (31)
- $v_b$  — wektor prędkości liniowej w układzie ciała (32)
- $v_s$  — wektor prędkości liniowej w układzie przestrzeni (32)
- $\mathcal{W}$  — przestrzeń robocza manipulatora (96)
- $X$  — pole wektorowe ruchu własnego (166)
- $X_{i_1 \dots i_{m+1}}$  — pole hamiltonowskie (136)
- $Y$  — macierz regresji (223)
- $Z$  — rozmaiłość zadaniowa (41)

**R**oger Bacon, którego cześć jako mistrza, nauczał nas, że Boski plan wskaże pewnego dnia wiedzę o machinach, które są magią naturalną i świętą. I będzie tak, że siły przyrody pozwolą zrobić maszyny do pływania, by okręty poruszały się tylko homine regente i znacznie szybciej niż te pchane wiatrem lub wiosłami; i będą wozy ut sine animalis moveantur cum impetu inestimabili, et instrumenta volandi et homo sedens in medio instrumentis revolvens aliquod ingenium per quod alae artificialiter composita aerem verberent, ad modum avis volantis. I małe narzędzia, które podnosić będą nieskończenie wielkie ciężary, i wozy, którymi jeździć będzie można po dnie morza.

U. Eco, *Imię róży*.

(tłum. A. Szymanowski)

# Rozdział 1

## Wprowadzenie

Wiele znaków wskazuje na to, że nadszedł czas „magii naturalnej i świętej”, który przepowiada Roger Bacon, zwany *doctor mirabilis*, franciszkanin, wynalazca okularów i konstruktor androidu, a imieniem owej magii jest robotyka. Robotyka traktuje o samoczynnych maszynach, które naśladują funkcje manualne i intelektualne człowieka. Samoczynne maszyny nazywamy automatami. Aby automat stał się przedmiotem zainteresowań robotyki, powinien mieć pewien stopień samodzielności, czyli zdolności do celowego funkcjonowania w różnorodnych, nie w pełni znanych, środowiskach. Własność samodzielności, a więc względnej niezależności od otoczenia, często nazywa się autonomią lub inteligencją. Ponieważ zapewnienie samodzielności automatu wymaga nieustannego czerpania i przetwarzania informacji o środowisku, definiuje się robotykę jako naukę o inteligentnym wykorzystaniu percepcji do działania. W tym kontekście, obiekty robotyki nazywane tradycyjnie robotami można określić jako samoczynne i samodzielne maszyny, autonomiczne automaty lub obdarzyć mianem inteligentnych agentów. Jest przeto robotyka nauką interdyscyplinarną, zachwycającą i budzącą przerażenie *ut castrorum acies ordinata*, zakorzenioną w takich dyscyplinach jak mechanika, elektronika i cybernetyka (w klasycznym, wienerowskim rozumieniu). Jak być powinno w każdej ścisłej nauce, językiem i narzędziem robotyki jest matematyka, albowiem w myśl słów Wilhelma z Baskerville:

*„Wiedza matematyczna składa się z twierdzeń zbudowanych przez nasz umysł w ten sposób, by zawsze funkcjonowały jako prawda, albo dlatego że są przyrodzone, albo dlatego że matematyka była wynaleziona w pierw niż inne nauki.”* (U. Eco, *Imię róży*)

Na fundamencie matematycznym opiera się wielka różnorodność me-

to i teorii, heurystyk i hipotez, technik i technologii robotyki. Niniejsza książka ogranicza się do niewielkiego fragmentu tej różnorodności, jakim są podstawy robotyki, i dostarcza opisu podstawowych problemów robotyki i sposobów ich rozwiązania (algorytmów) odnoszących się do modelowania kinematyki i dynamiki, planowania ruchu oraz sterowania manipulatorów i robotów mobilnych. Z punktu widzenia zastosowań, omawiane w książce metody i algorytmy odnoszą się do dwóch klas robotów: robotów przemysłowych oraz robotów usługowych. Nasze ujęcie problemów robotyki ma charakter selektywny i koncentruje się na tych zagadnieniach, do których sformułowania lub rozwiązania udało nam się wnieść pewien wkład oryginalny. Obszary tematyczne robotyki, którym poświęciliśmy szczególną uwagę są zatem następujące: modele kinematyki i dynamiki manipulatorów i nieholonomicznych robotów mobilnych, modelowanie osobliwości kinematyki i kryteria unikania osobliwości manipulatorów redundantnych, algorytmy kinematyki odwrotnej manipulatorów z osobliwościami, algorytmy sterowania manipulatorów sztywnych i manipulatorów o elastycznych przegubach przy ograniczonej znajomości modelu dynamiki, algorytmy planowania ruchu robotów nieholonomicznych wykorzystujące metody geometrycznej teorii sterowania, algorytmy sterowania kołowych robotów mobilnych. W wymienionych obszarach własne wyniki autorów książki zostały przedstawione w szerokim kontekście wyników udokumentowanych w literaturze, dając, jak sądzimy, wyczerpujący i aktualny obraz sytuacji problemowej w dziedzinie podstaw robotyki oraz otwierając nowe perspektywy, zarówno w zakresie badań teoretycznych, jak i stosowanych\*.

Książka opiera się na rezultatach badań naukowych, jakie prowadziliśmy w latach 1991–1999, w ramach trzech projektów badawczych finansowanych przez Komitet Badań Naukowych:

- Metody topologiczno-różniczkowe w robotyce,
- Roboty osobliwe i nieholonomiczne: modele i algorytmy sterowania,
- Roboty osobliwe i nieholonomiczne: modele, sterowanie i planowanie trajektorii,

jak również kilku grantów statutowych KBN. Wyniki szczegółowe zostały udokumentowane w rozprawie habilitacyjnej [Dul98], trzech rozprawach

---

\* Jakkolwiek podstawowym narzędziem weryfikacji wyników teoretycznych są w książce symulacje komputerowe, wypada zaznaczyć, że metoda postaci normalnych rozwiązania osobliwego odwrotnego zadania kinematyki oraz algorytmy sterowania manipulatorów sztywnych typu  $\lambda$ -śledzenia przeszły pomyślnie weryfikację eksperymentalną.

doktorskich [Hos96, Maz96, Mus96] oraz w kilkudziesięciu publikacjach. Oprócz wyników opublikowanych, nie oparliśmy się pokusie zamieszczenia w książce wyników zupełnie nowych, uzyskanych przy pracy nad projektem badawczym KBN pt. Modelowanie, planowanie ruchu i sterowanie manipulatorów mobilnych. Sądzymy, że prace których podsumowaniem jest ta książka stanowią dorobek pewnej szkoły uprawiania robotyki, jaka powstała w Zakładzie Podstaw Cybernetyki i Robotyki Instytutu Cybernetyki Technicznej Politechniki Wrocławskiej. Pragniemy ją zadedykować naszemu Mistrzowi, zmarłemu w 1991 roku **Profesorowi Jerzemu Jaroniowi**, którego fascynacja robotyką ukształtowała nasze zainteresowania badawcze, i którego arystokratycznej kulturze logiczno-matematycznej zawdzięczamy podstawy warsztatu naukowego, [Jar78]. Autorzy książki mają zaszczyt zaliczać się do dzieci i wnuków naukowych Profesora.

Książka składa się z trzech części:

- Kinematyka manipulatorów i robotów mobilnych,
- Algorytmy sterowania manipulatorów,
- Algorytmy planowania ruchu i sterowania robotów mobilnych,

i dzieli się na 11 rozdziałów oraz 3 dodatki zawierające definicje podstawowych pojęć matematycznych, ze szczególnym uwzględnieniem dziedziny stabilności układów dynamicznych. Wydaje się nam, że wiedza z analizy matematycznej, algebry i teorii równań różniczkowych w zakresie wykładanym inżynierom, na przykład na kierunku Automatyka i Robotyka politechnik, wsparta podstawowym kursem teorii sterowania i pewną otwartością matematyczną, jest wystarczająca do zrozumienia większości materiału przedstawionego w tej książce. W posługiwaniu się bardziej zaawansowanymi technikami matematycznymi staraliśmy się zachować pewną powściągliwość (dotyczy to zwłaszcza geometrii różniczkowej), aczkolwiek nie podzielamy obawy, że pojawienie się w tekście robotycznym nawiasu Liego uczyni ten tekst całkowicie niezrozumiałym. Co więcej, jesteśmy przekonani, że język matematyczny ułatwia komunikowanie problemów, metod i twierdzeń robotyki nie tylko robotykom, lecz także badaczom spoza kręgu profesjonalnych robotyków. Jakościowy postęp w zakresie metod planowania ruchu i sterowania robotów mobilnych, jaki miał miejsce w bieżącej dekadzie, zawdzięczamy sformułowaniu pewnych zagadnień robotyki w sposób interesujący dla matematyków.

Cechą wyróżniającą nasz sposób podejścia jest poszukiwanie równoważności, transformowanie złożonych modeli i zadań do jak najprostszej po-

staci równoważnej, umożliwiającej ich rozwiązanie. W perspektywie filozoficzno-metodologicznej takie podejście można wyprowadzić z poglądów W. Ockhama<sup>†</sup>. Źródła matematyczne metody postaci normalnych znajdują się u H. Poincaré'go i E. Cartana; punktem odniesienia dla naszego sposobu myślenia na temat równoważności są prace B. Jakubczyka [Jak90]. W zasadzie unikamy w książce przedstawiania dowodów twierdzeń odsyłając zainteresowanego Czytelnika do źródłowej literatury. Wyjątkiem są dwa standardowe dowody stabilności algorytmów sterowania, które zamieściliśmy w specjalnym dodatku. Dla wygody Czytelnika wywody formalne zostały zilustrowane licznymi przykładami obliczeniowymi, a także symulacjami komputerowymi przeprowadzonymi w środowiskach MATHEMATICA<sup>®‡</sup> i MATLAB<sup>®</sup> (SIMULINK<sup>®§</sup>). Każdy rozdział kończymy serią komentarzy i uwag bibliograficznych umożliwiających Czytelnikowi dotarcie do źródeł przedstawionych wyników i naświetlających dodatkowo ich genezę, znaczenie i wzajemne powiązania. Szybkie odnalezienie potrzebnych pojęć ułatwia indeks zamieszczony na końcu książki.

Różne fragmenty tej książki służyły jako podstawa do wykładów z przedmiotów Mechanika analityczna, Podstawy robotyki oraz Systemy sterowania robotów, jakie prowadzimy na kierunku Automatyka i Robotyka, na Wydziale Elektroniki Politechniki Wrocławskiej. Reakcje naszych studentów, którzy, najprawdopodobniej bez własnej winy, byli kształceni na kilku roboczych wersjach tekstu tej książki, staraliśmy się wykorzystać przy opracowaniu wersji ostatecznej. Jesteśmy wdzięczni tym studentom, dla których robotyka nie pozostała dziedziną obojętną. Są wśród nich nie tylko tacy, którzy z naszych wykładów wynieśli przeświadczenie, że „robotyka to koszmar”, lecz również tacy, którzy w czasie studiów, niekiedy jeszcze przed napisaniem pracy dyplomowej, uzyskali własne wyniki naukowe. Za pomoc przy wykonaniu symulacji komputerowych dziękujemy przedstawicielowi tej drugiej grupy, mgrowi inż. T. Wróblewskiemu.

Szczególłą wdzięczność chcemy wyrazić Redaktorowi Naukowemu Akademickiej Oficyny Wydawniczej, Panu Profesorowi L. Bolcowi, który odważył się podjąć ryzyko związane z publikacją tej książki i zechciał nam udzielić fachowych wskazówek edytorskich w trakcie przygotowania jej do druku.

---

<sup>†</sup>Nazwanego *venerabilis inceptor*, podobnie jak R. Bacon franciszkanina z Oksfordu, autora dyrektywy metodologicznej „*Entia non sunt multiplicanda praeter necessitatem*”, znanej jako brzytwa Ockhama.

<sup>‡</sup>MATHEMATICA<sup>®</sup> jest znakiem firmowym Wolfram Research, Inc.

<sup>§</sup>MATLAB<sup>®</sup> i SIMULINK<sup>®</sup> są znakami firmowymi The MathWorks, Inc.

Na zakończenie dziękujemy Recenzentowi naszej książki, Panu Profesorowi K. Kozłowskiemu, którego kompetencje merytoryczne i profesorska wnikliwość w znacznym stopniu wpłynęły na ostateczną formę i treść książki. Zamieszczone w recenzji stwierdzenie, że Recenzent dwukrotnie przeczytał maszynopis książki traktujemy jako dowód, że książka nadaje się do czytania. Mamy przy tym świadomość, że odpowiedzialność za błędy, które pomimo naszych wysiłków redakcyjnych pozostały w książce, spoczywa wyłącznie na jej autorach.

Jak już napisaliśmy, ze względu na swoją specyficzną genezę, nasza książka traktuje o podstawach robotyki i ze zrozumiałych względów nie pokrywa całego zakresu problemowego współczesnej robotyki. W kilku dziedzinach robotyki wykraczających całkowicie lub częściowo poza zakres tej książki były prowadzone w Polsce w latach 90-tych poważne prace badawcze, których wyniki zostały opublikowane w formie rozpraw habilitacyjnych, książek lub monografii, mogących stanowić tematyczne uzupełnienie tej książki. Wymienimy niektóre z nich. Pracą zbiorową o charakterze przekrojowym jest książka [MK99] pod redakcją A. Moreckiego i J. Knapczyka. Metody projektowania manipulatorów przemysłowych zostały przedstawione w monografii K. Tomaszewskiego [Tom93]. O identyfikacji modeli dynamiki robotów manipulacyjnych traktuje rozprawa habilitacyjna K. Kozłowskiego [Koz92] oraz monografia [Koz98]. Zagadnieniom planowania trajektorii manipulatorów została poświęcona monografia M. Galickiego [Gal00]. Problematyka syntezy chwytu, bliska w swym ujęciu kinetycznym teorii układów nieholonomicznych, jest obecna w rozprawie habilitacyjnej A. Kasińskiego [Kas98]. Modeli kinematyki, dynamiki i algorytmów sterowania robotów mobilnych dotyczą rozprawy habilitacyjne Z. Hendzla [Hen96] i W. Żylskiego [Żyl96]. Planowaniem trajektorii kołowych robotów mobilnych zajął się w monografii habilitacyjnej L. Podśędkowski [Pod99]. Wprowadzenie do inteligentnych układów robotycznych stanowi rozprawa habilitacyjna W. Jacaka [Jac91] i monografia [Jac99]. Zagadnień modelowania elastycznych systemów produkcyjnych dotyczy książka Z. Banaszaka [BJ91]. Metod programowania robotów zostały poświęcone rozprawy habilitacyjne C. Zielińskiego [Zie95b] i T. Kocha [Koc96]. Zagadnienia syntezy ruchu maszyn kroczących były przedmiotem rozprawy habilitacyjnej T. Zielińskiej [Zie95a]. Aktywności Wydawnictw Naukowo-Technicznych zawdzięczamy publikację przekładów książek [Cra93, SV97]. Aktualny przegląd wyników badań prowadzonych w Polsce w zakresie robotyki zawierają materiały Krajowych Konferencji Robotyki.

## Literatura

- [BJ91] Z. Banaszak i L. S. Jampolski, *Komputerowo wspomagane modelowanie elastycznych systemów produkcyjnych*. WNT, Warszawa, 1991.
- [Cra93] J. J. Craig, *Wprowadzenie do robotyki*. WNT, Warszawa, 1993.
- [Dul98] I. Dulęba, *Algorithms of Motion Planning for Nonholonomic Robots*. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 1998.
- [Gal00] M. Galicki, *Wybrane metody planowania optymalnych trajektorii robotów manipulacyjnych*. WNT, Warszawa, 2000.
- [Hen96] Z. Hendzel, *Sterowanie ruchem nadźnym mobilnych robotów kołowych*. Oficyna Wydawnicza Politechniki Rzeszowskiej, Rzeszów, 1996.
- [Hos96] R. Hossa, *Modele i algorytmy sterowania kołowych robotów mobilnych*. Rozprawa doktorska, Instytut Cybernetyki Technicznej, Politechnika Wrocławska, 1996.
- [Jac91] W. Jacak, *Roboty inteligentne*. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 1991.
- [Jac99] W. Jacak, *Intelligent Robotic Systems*. Kluwer, New York, 1999.
- [Jak90] B. Jakubczyk, Equivalence and invariants of nonlinear control systems. W: H. J. Sussmann, (red.), *Differential Geometric Control Theory*, strony 177–218. M. Dekker, New York, 1990.
- [Jar78] J. Jaroń, *Systemic Prolegomena to Theoretical Cybernetics*. Wydawnictwo Politechniki Wrocławskiej, Wrocław, 1978.
- [Kas98] A. Kasiński, *Metody syntezy chwytu dla autonomicznych systemów manipulacyjnych*. Oficyna Wydawnicza Politechniki Poznańskiej, Poznań, 1998.
- [Koc96] T. Koch, *Programowanie redundantnych robotów przemysłowych*. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 1996.
- [Koz92] K. Kozłowski, *Modele matematyczne dynamiki robotów oraz identyfikacja parametrów tych modeli*. Oficyna Wydawnicza Politechniki Poznańskiej, Poznań, 1992.
- [Koz98] K. Kozłowski, *Modelling and Identification in Robotics*. Springer-Verlag, Berlin, 1998.
- [Maz96] A. Mazur, *Algorytmy sterowania robotów oparte na zasadzie uniwersalnego adaptacyjnego układu sterowania*. Rozprawa doktorska, Instytut Cybernetyki Technicznej, Politechnika Wrocławska, 1996.



- [MK99] A. Morecki i J. Knapczyk, (red.), *Podstawy Robotyki. Teoria i Elementy Manipulatorów*. WNT, Warszawa, 1999.
- [Mus96] R. Muszyński, *Modele i algorytmy sterowania manipulatorów z osobliwościami kinematycznymi*. Rozprawa doktorska, Instytut Cybernetyki Technicznej, Politechnika Wrocławska, 1996.
- [Pod99] L. Podsędkowski, *Dynamiczne planowanie trajektorii robotów mobilnych w zmiennej przestrzeni roboczej*. Wydawnictwo Politechniki Łódzkiej, Łódź, 1999.
- [SV97] M. Spong i M. Vidyasagar, *Dynamika i sterowanie robotów*. WNT, Warszawa, 1997.
- [Tom93] K. Tomaszewski, *Roboty przemysłowe*. WNT, Warszawa, 1993.
- [Zie95a] T. Zielińska, *Wykorzystanie własności chodu człowieka i zwierząt do syntezy ruchu maszyn kroczących*. Wydawnictwo Instytutu Biocybernetyki i Inżynierii Biomedycznej PAN, Warszawa, 1995.
- [Zie95b] C. Zieliński, *Robot Programming Methods*. Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa, 1995.
- [Żyl96] W. Żylski, *Kinematyka i dynamika mobilnych robotów kołowych*. Oficyna Wydawnicza Politechniki Rzeszowskiej, Rzeszów, 1996.

Do składu książki wykorzystano system przygotowania dokumentów  $\text{\LaTeX}$ , opracowany przez L. Lamporta [Lam94], będący nakładką systemu  $\text{\TeX}$ , [Knu86a, Knu86b]. Matematyczne czcionki o nazwie AMS Euler, których używamy w tej książce, zostały opracowane przez H. Zapfa [KZ86] na zlecenie Amerykańskiego Towarzystwa Matematycznego. Czcionki składu tekstu, zwane Concrete Roman i Concrete Italic, należące do knuthowskiej rodziny czcionek Computer Modern, zostały specjalnie przystosowane do kształtu czcionki AMS Euler na potrzeby książki [GKP96]. Wszystkie symulacje komputerowe zamieszczone w książce zostały przeprowadzone w środowiskach obliczeniowych MATHEMATICA<sup>®</sup> firmy Wolfram Research, Inc., [Wol96], oraz MATLAB<sup>®</sup> i SIMULINK<sup>®</sup> firmy The MathWorks, Inc., [Mat94a, Mat94b].

- [GKP96] R. L. Graham, D. E. Knuth i O. Patashnik, *Matematyka konkretna*. PWN, Warszawa, 1996.
- [Knu86a] D. E. Knuth, *The  $\text{\TeX}$ book*, volume A of Computers and Typesetting. Addison-Wesley, Reading, 1986.
- [Knu86b] D. E. Knuth,  *$\text{\TeX}$ : The Program*, volume B of Computers and Typesetting. Addison-Wesley, Reading, 1986.
- [KZ86] D. E. Knuth i H. Zapf, *AMS Euler — A new typeface for mathematics*. Scholarly Publishing, 20:131–157, 1986.
- [Lam94] L. Lamport,  *$\text{\LaTeX}$ : A Document Preparation System*. Addison-Wesley, Reading, 1994.
- [Mat94a] The MathWorks, Inc., *MATLAB User's Guide*, 1994.
- [Mat94b] The MathWorks, Inc., *SIMULINK User's Guide*, 1994.
- [Wol96] S. Wolfram, *The MATHEMATICA<sup>®</sup> Book*. Cambridge University Press, Cambridge, 1996.



Część I

**Kinematyka manipulatorów  
i robotów mobilnych**



## Rozdział 2

# Modele kinematyki układów robotycznych

### 2.1 Ruch ciała sztywnego

Wszystkie obiekty robotyki istnieją w czasoprzestrzeni fizycznej złożonej z jednowymiarowego czasu i trójwymiarowej przestrzeni. *Czas fizyczny* jest zbiorem chwil, który będziemy utożsamiać ze zbiorem liczb rzeczywistych  $\mathbb{R}$ . *Przestrzeń fizyczna* składa się z punktów, których położenie względem zadanego, prawoskrętnego układu współrzędnych kartezjańskich (zwanego układem przestrzeni) można wyrazić przy pomocy trójki liczb rzeczywistych. W efekcie, przestrzeń fizyczną utożsamiamy z przestrzenią *euklidesową*  $\mathbb{R}^3$  wyposażoną w operacje *mnożenia skalarnego* i *mnożenia wektorowego*. Weźmy dwa wektory  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^3$ ,  $\mathbf{x} = (x_1, x_2, x_3)^T$ ,  $\mathbf{y} = (y_1, y_2, y_3)^T$  i niech wektory  $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$  stanowią standardową bazę w  $\mathbb{R}^3$ . Iloczyn skalarny i iloczyn wektorowy definiujemy następująco

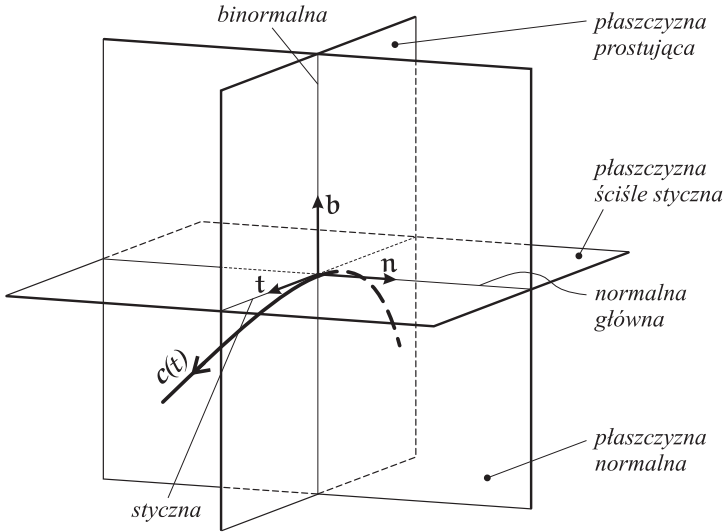
$$\mathbf{x} \cdot \mathbf{y} = \mathbf{x}^T \mathbf{y} = \langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^3 x_i y_i, \quad \mathbf{x} \times \mathbf{y} = \det \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix}. \quad (2.1)$$

Miarę *długości* wektora  $\mathbf{x} \in \mathbb{R}^3$  w przestrzeni euklidesowej wyznacza norma euklidesowa  $\|\mathbf{x}\| = \sqrt{\mathbf{x} \cdot \mathbf{x}}$ . Każde gładkie\* przekształcenie czasu w przestrzeń

$$\mathbf{c} : \mathbb{R} \longrightarrow \mathbb{R}^3 \quad (2.2)$$

---

\*Klasy  $C^\infty$ , tzn. posiadające ciągłe pochodne dowolnego rzędu.



Rysunek 2.1 Trójścian Freneta.

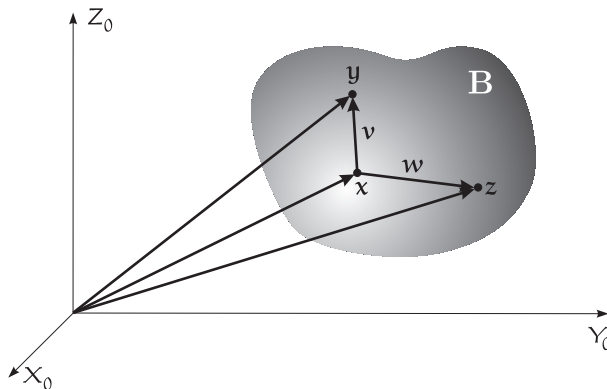
nazywamy *ruchem* punktu materialnego. Przestrzeń  $\mathbb{R}^3$  nazywamy *przestrzenią konfiguracyjną* punktu materialnego. Wykres ruchu (2.2) w czasoprzestrzeni stanowi *trajektorię* ruchu. Krzywą w  $\mathbb{R}^3$  będącą obrazem ruchu nazywamy *torem* (*ścieżką*) ruchu. Pochodną  $\dot{\mathbf{c}}(t)$  ruchu względem czasu nazywamy *prędkością* ruchu. Wektor prędkości jest styczny do toru ruchu. Pochodną rzędu drugiego  $\ddot{\mathbf{c}}(t)$  ruchu względem czasu nazywamy *przyspieszeniem* ruchu. Lokalnie geometrię toru ruchu opisuje tzw. *trójścian Freneta*, przedstawiony na rysunku 2.1. Trójścian Freneta jest rozpięty przez wektory jednostkowe (wersory)  $\mathbf{t}$ ,  $\mathbf{n}$ ,  $\mathbf{b}$ , zdefiniowane w następujący sposób. Wektor styczny  $\mathbf{t} = \frac{\dot{\mathbf{c}}}{\|\dot{\mathbf{c}}\|}$ . Wektor normalny  $\mathbf{n} = \frac{1}{K} \frac{d\mathbf{t}}{ds}$ , gdzie  $ds = \|\dot{\mathbf{c}}\| dt$  jest elementem długości toru, natomiast

$$K = \left\| \frac{d\mathbf{t}}{ds} \right\| = \frac{\sqrt{\|\dot{\mathbf{c}}\|^2 \|\ddot{\mathbf{c}}\|^2 - (\dot{\mathbf{c}} \cdot \ddot{\mathbf{c}})^2}}{\|\dot{\mathbf{c}}\|^3}$$

nosi nazwę *krzywizny* toru i stanowi miarę odchylenia toru od linii prostej. Wektor binormalny  $\mathbf{b}$  jest iloczynem wektorowym dwóch poprzednio zdefiniowanych wektorów

$$\mathbf{b} = \mathbf{t} \times \mathbf{n} = \frac{1}{K \|\dot{\mathbf{c}}\|^3} (\dot{\mathbf{c}} \times \ddot{\mathbf{c}}).$$

Odchylenie toru od krzywej płaskiej określa parametr zwany *skręceniem*

Rysunek 2.2 Ciało sztywne w  $\mathbb{R}^3$ .

lub torsją toru

$$T = \left\| \frac{d\mathbf{b}}{ds} \right\| = \frac{1}{K^2} \frac{|(\dot{\mathbf{c}} \times \ddot{\mathbf{c}}) \cdot \dot{\mathbf{c}}|}{\|\dot{\mathbf{c}}\|^6}.$$

Występujący w powyższej formule *iloczyn mieszany* wektorów  $\mathbf{x} = (x_1, x_2, x_3)^T$ ,  $\mathbf{y} = (y_1, y_2, y_3)^T$ ,  $\mathbf{z} = (z_1, z_2, z_3)^T$  definiuje się jako

$$(\mathbf{x} \times \mathbf{y}) \cdot \mathbf{z} = \det \begin{bmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \end{bmatrix}.$$

Nietrudno się przekonać, że wektor przyspieszenia  $\ddot{\mathbf{c}}(t)$  leży w płaszczyźnie stycznej do toru i w ogólnym przypadku posiada dwie składowe: styczną i normalną.

Rozważmy afiniczne przekształcenie przestrzeni euklidesowej

$$\mathbf{D} : \mathbb{R}^3 \longrightarrow \mathbb{R}^3, \quad \mathbf{D}(\mathbf{x}) = \mathbf{R}\mathbf{x} + \mathbf{T}, \quad (2.3)$$

gdzie  $\mathbf{R}$  jest macierzą rozmiaru  $3 \times 3$ , zaś wektor  $\mathbf{T} \in \mathbb{R}^3$ . Niech zwarty (domknięty i ograniczony) podzbiór  $B \subset \mathbb{R}^3$  oznacza *ciało sztywne* (rysunek 2.2). Wybierzmy punkty  $\mathbf{x}, \mathbf{y}, \mathbf{z} \in B$  i zdefiniujmy wektory swobodne  $\mathbf{v} = \mathbf{y} - \mathbf{x}$ ,  $\mathbf{w} = \mathbf{z} - \mathbf{x}$  łączące punkt  $\mathbf{x}$  z punktami  $\mathbf{y}$  i  $\mathbf{z}$ . Przekształcenie (2.3) można w naturalny sposób przenieść na wektory swobodne określając

$$\mathbf{D}_*(\mathbf{v}) = \mathbf{D}(\mathbf{y}) - \mathbf{D}(\mathbf{x}) = \mathbf{R}\mathbf{v}, \quad \mathbf{D}_*(\mathbf{w}) = \mathbf{D}(\mathbf{z}) - \mathbf{D}(\mathbf{x}) = \mathbf{R}\mathbf{w}. \quad (2.4)$$

Przekształcenie afiniczne opisane wzorem (2.3) nazywamy *przemieszczeniem* ciała sztywnego w przestrzeni euklidesowej, jeżeli  $\mathbf{D}_*$  zachowuje iloczyn skalarny i iloczyn wektorowy<sup>†</sup>, tzn.

$$\mathbf{D}_*(\mathbf{v}) \cdot \mathbf{D}_*(\mathbf{w}) = \mathbf{v} \cdot \mathbf{w} \quad \text{oraz} \quad \mathbf{D}_*(\mathbf{v} \times \mathbf{w}) = \mathbf{D}_*(\mathbf{v}) \times \mathbf{D}_*(\mathbf{w}). \quad (2.5)$$

Na mocy definicji, przemieszczenie ciała sztywnego zachowuje długość wektorów swobodnych (odległość między punktami) oraz kąt między wektorami swobodnymi

$$\|\mathbf{D}_*(\mathbf{v})\| = \|\mathbf{v}\|, \quad \angle(\mathbf{D}_*(\mathbf{v}), \mathbf{D}_*(\mathbf{w})) = \angle(\mathbf{v}, \mathbf{w}).$$

Biorąc pod uwagę definicję (2.4) i własności (2.5), nietrudno otrzymać następującą charakterystykę przemieszczenia ciała sztywnego w przestrzeni euklidesowej:

- macierz  $\mathbf{R}$  jest ortogonalna:  $\mathbf{R}\mathbf{R}^\top = \mathbf{R}^\top\mathbf{R} = \mathbb{I}_3$ ,
- $\mathbf{R}(\mathbf{v} \times \mathbf{w}) = (\mathbf{R}\mathbf{v}) \times (\mathbf{R}\mathbf{w})$ ,
- $\det \mathbf{R} = ((\mathbf{R}\mathbf{e}_1) \times (\mathbf{R}\mathbf{e}_2)) \cdot (\mathbf{R}\mathbf{e}_3) = \|\mathbf{R}\mathbf{e}_3\|^2 = 1^\ddagger$ ,
- wektor  $\mathbf{T}$  jest dowolny.

Macierze  $\mathbf{R}$  spełniające pierwsze trzy warunki nazywają się *macierzami obrotu* i tworzą *specjalną grupę obrotów*  $\mathbb{S}\mathbb{O}(3)$ .

Dzięki wprowadzeniu do przestrzeni euklidesowej tzw. *współrzędnych jednorodnych* pozwalających opisać punkt o współrzędnych  $\mathbf{x} = (x_1, x_2, x_3)^\top$  układem czterech liczb  $(x_1, x_2, x_3, 1)^\top$ , można przedstawić przekształcenie afiniczne (2.3) przy pomocy macierzy rozmiaru  $4 \times 4$  postaci  $\begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix}$ . Wówczas

$$\begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} \mapsto \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix} \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R}\mathbf{x} + \mathbf{T} \\ 1 \end{pmatrix}. \quad (2.6)$$

Transformacja (2.6) posiada dwojaką interpretację. Po pierwsze, opisuje ona zmianę współrzędnych jednorodnych  $(\mathbf{x}^\top, 1)^\top$  (przemieszczenie) pewnego punktu w przestrzeni euklidesowej względem ustalonego układu przestrzeni. Po drugie, jeżeli punkt  $(\mathbf{x}^\top, 1)^\top$  jest ustalony, transformacja (2.6) charakteryzuje przemieszczenie pewnego układu współrzędnych względem

<sup>†</sup>Ściśle mówiąc, ze wzoru (2.5) wynika, że iloczyn skalarny jest inwariantny (niezmienny), natomiast iloczyn wektorowy jest ekwiwariantny (równozmienny) względem przemieszczenia ciała sztywnego.

<sup>‡</sup>Wykorzystujemy tu prawoskrętność układu  $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ .



układu przestrzeni, które zachodzi w taki sposób, że jeżeli  $(\mathbf{x}^T, 1)^T$  oznacza współrzędne jednorodnie ustalonego punktu w układzie przemieszczonym, to  $((\mathbf{R}\mathbf{x} + \mathbf{T})^T, 1)^T$  określa współrzędne jednorodnie tego punktu w układzie przestrzeni. Korzystając z tej drugiej interpretacji rozważmy przemieszczenie ciała sztywnego w przestrzeni euklidesowej. Ponieważ podczas przemieszczania odległość między punktami ciała sztywnego nie ulega zmianie, jego ruch jest zdeterminowany przez ruch dowolnie wybranego punktu ciała. Wybierzmy taki punkt i umieśćmy w nim prawoskrętny, kartezjański układ współrzędnych związany z ciałem sztywnym (układ ciała). Załóżmy, że w chwili początkowej układ ciała i układ przestrzeni pokrywają się. Po przemieszczeniu elementy  $\mathbf{R}$ ,  $\mathbf{T}$  przekształcenia afinicznego (2.3) wyznaczają położenie ( $\mathbf{T}$ ) i orientację ( $\mathbf{R}$ ) układu ciała względem układu przestrzeni. W konsekwencji, ruch ciała sztywnego w przestrzeni euklidesowej może być rozumiany jako gładkie przekształcenie czasu w grupę przesunięć  $\mathbb{R}^3$  oraz w specjalną grupę obrotów  $\mathbb{SO}(3)$ ,

$$\mathbf{c} : \mathbb{R} \longrightarrow \mathbb{R}^3 \times \mathbb{SO}(3) \cong \mathbb{SE}(3),$$

które określa w każdej chwili położenie i orientację układu ciała względem układu przestrzeni. Występujący w powyższym wzorze obiekt  $\mathbb{SE}(3)$  nazywa się *specjalną grupą euklidesową* i stanowi *przestrzeń konfiguracyjną* ciała sztywnego. Zauważmy, że zależność (2.6) pozwala na reprezentowanie elementów specjalnej grupy euklidesowej przy pomocy macierzy  $\begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix}$ . Tę macierzową reprezentację grupy  $\mathbb{SE}(3)$  będziemy odtąd utożsamiać z grupą  $\mathbb{SE}(3)$ . Nietrudno wykazać, że  $\mathbb{SE}(3)$  istotnie jest grupą z działaniem grupowym będącym mnożeniem macierzy

$$\begin{bmatrix} \mathbf{R}_1 & \mathbf{T}_1 \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_2 & \mathbf{T}_2 \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1\mathbf{R}_2 & \mathbf{R}_1\mathbf{T}_2 + \mathbf{T}_1 \\ \mathbf{0} & 1 \end{bmatrix}, \quad (2.7)$$

elementem neutralnym  $\mathbb{E} = \mathbb{I}_4$  oraz elementem odwrotnym

$$\begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{R}^T & -\mathbf{R}^T\mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix}.$$

Jak wynika ze wzoru (2.7), specjalna grupa euklidesowa jest tzw. iloczynem półprostym dwóch grup: podgrupy przesunięć  $\mathbb{R}^3$  i podgrupy obrotów  $\mathbb{SO}(3)$ . Zauważmy, że na mocy definicji każdy element  $\mathbf{S} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{SE}(3)$ , formalnie należący do przestrzeni  $\mathbb{R}^{12}$  ( $\mathbf{R} \in \mathbb{R}^9$ ,  $\mathbf{T} \in \mathbb{R}^3$ ), spełnia 6 niezależnych warunków ortogonalności  $\mathbf{R}^T\mathbf{R} = \mathbb{I}_3$ , co pozostawia elementom macierzy  $\mathbf{S}$

sześć stopni swobody. Z tego powodu powiadamy, że specjalna grupa euklidesowa ma wymiar 6,  $\dim \text{SE}(3) = 6$ . Grupa  $\text{SE}(3)$  stanowi przykład obiektu matematycznego zwanego *gładką rozmaitością*. W dalszym ciągu przez gładką rozmaitość *kowymiaru*  $k$  będziemy rozumieć podzbiór pewnej przestrzeni  $\mathbb{R}^n$  zdefiniowany przy pomocy  $k$  warunków (więzów, ograniczeń) postaci

$$\mathcal{M} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{f}(\mathbf{x}) = \mathbf{0}\},$$

gdzie  $\mathbf{f} = (f_1, f_2, \dots, f_k)^\top$ , a funkcje  $f_1, \dots, f_k$  są gładkie (klasy  $C^\infty$ ) oraz niezależne w każdym punkcie  $\mathbf{x} \in \mathcal{M}$ , to znaczy rząd macierzy Jacobiego ograniczeń

$$\text{rank} \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] (\mathbf{x}) = k.$$

Liczbę  $m = n - k$  nazywamy *wymiarem* rozmaitości  $\mathcal{M}$ . W każdym punkcie  $\mathbf{x} \in \mathcal{M}$  gładkiej rozmaitości jest określona tzw. *przestrzeń styczna*,

$$\mathbf{T}_{\mathbf{x}}\mathcal{M} = \text{Ker} \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right] (\mathbf{x}),$$

złożona z wektorów anihilowanych przez macierz Jacobiego ograniczeń w tym punkcie. Na mocy definicji rozmaitości,  $\dim \mathbf{T}_{\mathbf{x}}\mathcal{M} = \dim \mathcal{M} = m$ . Obiekt powstały ze sklejenia przestrzeni stycznych w poszczególnych punktach  $\mathbf{x} \in \mathcal{M}$ ,

$$\mathbf{T}\mathcal{M} = \bigcup_{\mathbf{x} \in \mathcal{M}} (\mathbf{x}, \mathbf{T}_{\mathbf{x}}\mathcal{M}),$$

nazywa się *wiązką styczną* rozmaitości  $\mathcal{M}$ .

Każda gładka rozmaitość  $m$ -wymiarowa jest lokalnie równoważna przestrzeni  $\mathbb{R}^m$ . Rozumiemy przez to, że istnieją wzajemnie jednoznaczne i gładkie przekształcenia

$$\varphi_{\mathbf{u}} : \mathbf{U} \longrightarrow \mathbf{V} \quad \text{oraz} \quad \psi_{\mathbf{v}} : \mathbf{V} \longrightarrow \mathbf{U},$$

określone na otwartych podzbiórach  $\mathbf{U} \subset \mathcal{M}$ ,  $\mathbf{V} \subset \mathbb{R}^m$ , będące wzajemnymi odwrotnościami<sup>§</sup> i zwane, odpowiednio, *układem współrzędnych* oraz *parametryzacją* rozmaitości  $\mathcal{M}$ . Para  $(\mathbf{U}, \varphi_{\mathbf{u}})$  nazywa się *mapą* rozmaitości  $\mathcal{M}$ .

<sup>§</sup>Tzn.  $\varphi_{\mathbf{u}} \circ \psi_{\mathbf{v}} = \text{id}_{\mathbf{V}}$ ,  $\psi_{\mathbf{v}} \circ \varphi_{\mathbf{u}} = \text{id}_{\mathbf{U}}$ .

Największy układ zgodnych map<sup>¶</sup>, którego dziedziny  $\mathbf{U}$  pokrywają  $\mathcal{M}$  nazywa się *atlasem* rozmaitości  $\mathcal{M}$ . Jak już powiedzieliśmy, specjalna grupa euklidesowa  $\mathbb{SE}(3)$  jest gładką (ściślej: analityczną) rozmaitością wymiaru 6 i jednocześnie grupą. Grupa posiadająca strukturę gładkiej rozmaitości, której działanie grupowe jest gładką funkcją współrzędnych, nosi nazwę *grupy Liego*.

Podobnie, jak w przypadku ruchu punktu materialnego, wprowadzimy obecnie pojęcie prędkości ruchu ciała sztywnego. W tym celu dla ruchu  $\mathbf{c}(t) = \begin{bmatrix} \mathbf{R}(t) & \mathbf{T}(t) \\ \mathbf{0} & 1 \end{bmatrix}$  obliczamy pochodną względem czasu  $\dot{\mathbf{c}}(t) = \begin{bmatrix} \dot{\mathbf{R}}(t) & \dot{\mathbf{T}}(t) \\ \mathbf{0} & 0 \end{bmatrix}$ . W naturalny sposób definiuje się dwa rodzaje prędkości ciała sztywnego: *prędkość w układzie przestrzeni*

$$\mathbf{V}^s = \dot{\mathbf{c}}(t)\mathbf{c}^{-1}(t) = \begin{bmatrix} \boldsymbol{\Omega}_s & \dot{\mathbf{T}} - \boldsymbol{\Omega}_s\mathbf{T} \\ \mathbf{0} & 0 \end{bmatrix} \quad (2.8)$$

oraz *prędkość w układzie ciała*

$$\mathbf{V}^b = \mathbf{c}^{-1}(t)\dot{\mathbf{c}}(t) = \begin{bmatrix} \boldsymbol{\Omega}_b & \mathbf{R}^T\dot{\mathbf{T}} \\ \mathbf{0} & 0 \end{bmatrix}. \quad (2.9)$$

W powyższych wyrażeniach  $\boldsymbol{\Omega}_s = \dot{\mathbf{R}}\mathbf{R}^T$  definiujemy jako *prędkość kątową w układzie przestrzeni*,  $\boldsymbol{\Omega}_b = \mathbf{R}^T\dot{\mathbf{R}}$  jest *prędkością kątową w układzie ciała*. Z ortogonalności macierzy obrotu  $\mathbf{R}^T\mathbf{R} = \mathbf{R}\mathbf{R}^T = \mathbb{I}_3$  wynika, że każda z macierzy  $\boldsymbol{\Omega}_s$ ,  $\boldsymbol{\Omega}_b$  jest skośnie symetryczna rozmiaru  $3 \times 3$ , a zatem zdeteterminowana przez wektory prędkości kątowej w układzie przestrzeni  $\boldsymbol{\omega}_s$  i w układzie ciała  $\boldsymbol{\omega}_b$ . Związek między macierzowymi a wektorowymi prędkościami kątowymi jest opisany przy pomocy odwzorowania

$$[\ ]: \mathbb{R}^3 \longrightarrow \text{macierze skośnie symetryczne rozmiaru } 3 \times 3,$$

zdefiniowanego formułą

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} = \boldsymbol{\omega} \longmapsto [\boldsymbol{\omega}] = \boldsymbol{\Omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \quad (2.10)$$

Wprowadzone odwzorowanie jest wzajemnie jednoznaczne i określa wektorową reprezentację prędkości w układzie przestrzeni i w układzie ciała

$$\boldsymbol{\Omega}_s = [\boldsymbol{\omega}_s], \quad \boldsymbol{\Omega}_b = [\boldsymbol{\omega}_b]. \quad (2.11)$$

---

<sup>¶</sup>Dwie mapy  $(\mathbf{U}_1, \boldsymbol{\varphi}_{\mathbf{U}_1})$ ,  $(\mathbf{U}_2, \boldsymbol{\varphi}_{\mathbf{U}_2})$  nazywamy *zgodnymi*, jeżeli złożenie  $\boldsymbol{\varphi}_{\mathbf{U}_2} \circ \boldsymbol{\varphi}_{\mathbf{U}_1}^{-1}$  jest lokalnym homeomorfizmem (zobacz dodatek A.2) przestrzeni  $\mathbb{R}^m$ .

Ponadto, dla każdego wektora  $\mathbf{v} \in \mathbb{R}^3$  jest spełniona zależność

$$\boldsymbol{\Omega}\mathbf{v} = [\boldsymbol{\omega}]\mathbf{v} = \boldsymbol{\omega} \times \mathbf{v}.$$

Reprezentacja (2.11) umożliwia zastąpienie macierzy prędkości  $\mathbf{V}^s, \mathbf{V}^b$  wektorami postaci

$$\mathbf{v}^s = \begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix}, \quad \mathbf{v}^b = \begin{pmatrix} \mathbf{v}_b \\ \boldsymbol{\omega}_b \end{pmatrix}, \quad (2.12)$$

gdzie

$$\mathbf{v}_s = \dot{\mathbf{T}} + \mathbf{T} \times \boldsymbol{\omega}_s, \quad \mathbf{v}_b = \mathbf{R}^T \dot{\mathbf{T}}$$

oznaczają, odpowiednio, *prędkość liniową w układzie przestrzeni i w układzie ciała*. Macierze  $\mathbf{V}^s, \mathbf{V}^b$  należą do przestrzeni stycznej do grupy  $\mathbb{SE}(3)$  w elemencie jednostkowym  $\mathbb{E} = \mathbb{I}_4$ , identyfikowanej z tzw. algebrą Liego  $\mathfrak{se}(3)$  grupy  $\mathbb{SE}(3)$ . Z zależności (2.12) wynika, że  $\mathfrak{se}(3)$  można utożsamić z przestrzenią  $\mathbb{R}^6$ .

Wprowadzonym w sposób formalny pojęciem prędkości ciała sztywnego w układzie przestrzeni i w układzie ciała można nadać następującą interpretację. Wyobraźmy sobie, że układ ciała przemieszcza się względem układu przestrzeni wzdłuż trajektorii  $\mathbf{c}(t) = \begin{bmatrix} \mathbf{R}(t) & \mathbf{T}(t) \\ \mathbf{0} & 1 \end{bmatrix}$ . Wybierzmy w układzie ciała pewien punkt  $\mathbf{P}$  o współrzędnych jednorodnych  $(\mathbf{p}^T, 1)^T$ . W chwili  $t$  współrzędne jednorodne tego punktu względem układu przestrzeni wynoszą

$$\begin{pmatrix} \mathbf{r}(t) \\ 1 \end{pmatrix} = \begin{bmatrix} \mathbf{R}(t) & \mathbf{T}(t) \\ \mathbf{0} & 1 \end{bmatrix} \begin{pmatrix} \mathbf{p} \\ 1 \end{pmatrix}. \quad (2.13)$$

Prędkość ruchu punktu  $\mathbf{P}$  względem układu przestrzeni

$$\begin{pmatrix} \dot{\mathbf{r}}(t) \\ 0 \end{pmatrix} = \begin{bmatrix} \dot{\mathbf{R}}(t) & \dot{\mathbf{T}}(t) \\ \mathbf{0} & 0 \end{bmatrix} \begin{pmatrix} \mathbf{p} \\ 1 \end{pmatrix} \quad (2.14)$$

wyrażona w układzie ciała jest równa

$$\begin{bmatrix} \mathbf{R}(t) & \mathbf{T}(t) \\ \mathbf{0} & 1 \end{bmatrix}^{-1} \begin{bmatrix} \dot{\mathbf{R}}(t) & \dot{\mathbf{T}}(t) \\ \mathbf{0} & 0 \end{bmatrix} \begin{pmatrix} \mathbf{p} \\ 1 \end{pmatrix} = \mathbf{V}^b \begin{pmatrix} \mathbf{p} \\ 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{\omega}_b \times \mathbf{p} + \mathbf{R}^T \dot{\mathbf{T}} \\ 0 \end{pmatrix}. \quad (2.15)$$

Z zależności (2.15) wynika, że prędkość w układzie ciała jest prędkością ruchu względem układu przestrzeni punktu  $\mathbf{P}$ , którego współrzędne w układzie

ciała są ustalone, widzianą z układu ciała. Z kolei, jeżeli chcemy uzależnić prędkość ruchu punktu  $\mathbf{P}$ , widzianego z układu przestrzeni, od współrzędnych jednorodnych tego punktu, z połączenia wzorów (2.13) i (2.14) otrzymujemy

$$\begin{aligned} \begin{pmatrix} \dot{\mathbf{r}}(t) \\ 0 \end{pmatrix} &= \begin{bmatrix} \dot{\mathbf{R}}(t) & \dot{\mathbf{T}}(t) \\ \mathbf{0} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R}(t) & \mathbf{T}(t) \\ \mathbf{0} & 1 \end{bmatrix}^{-1} \begin{pmatrix} \mathbf{r}(t) \\ 1 \end{pmatrix} = \\ &= \mathbf{V}^s \begin{pmatrix} \mathbf{r}(t) \\ 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{\omega}_s \times (\mathbf{r}(t) - \mathbf{T}(t)) + \dot{\mathbf{T}}(t) \\ 0 \end{pmatrix}. \end{aligned} \quad (2.16)$$

Formuła (2.16) oznacza, że prędkość punktu  $\mathbf{P}$  w układzie przestrzeni posiada składową pochodzącą od obrotu wektora  $\mathbf{r} - \mathbf{T}$  z prędkością  $\boldsymbol{\omega}_s$  oraz składową wynikającą z ruchu początku układu ciała. Mimo formalnego podobieństwa wyrażeń (2.15) i (2.16) należy pamiętać, że faktycznie punkt  $\mathbf{P}$  nie porusza się względem układu ciała.

Grupa  $\text{SE}(3)$ , traktowana jako gładka rozmaiłość, dopuszcza rozmaite układy współrzędnych i parametryzacje. Najczęściej przy ich konstruowaniu wykorzystuje się utożsamienie rozmaiłości  $\text{SE}(3) \cong \mathbb{R}^3 \times \text{SO}(3)$  i definiuje układ współrzędnych bądź parametryzację osobno dla podgrupy przesunięć  $\mathbb{R}^3$ , osobno dla podgrupy obrotów  $\text{SO}(3)$ . Trywialna parametryzacja w podgrupie przesunięć jest wyznaczona przez przekształcenie tożsamościowe  $\boldsymbol{\psi}_{\mathbf{V}}(\mathbf{T}) = \mathbf{T}$  wprowadzające *kartezjański* układ współrzędnych. Oczywiście, w tym przypadku  $\mathbf{V} = \mathbb{R}^3$ . Współrzędne *walcowe* i *sferyczne* pochodzą od parametryzacji grupy przesunięć opisanych następującymi formułami<sup>||</sup>

$$\begin{aligned} \boldsymbol{\psi}_{\mathbf{V}}(r, \varphi, z) &= (r \cos \varphi, r \sin \varphi, z)^T = \mathbf{T}, \\ \mathbf{V} &= \left\{ (r, \varphi, z)^T \in \mathbb{R}^3 \mid r > 0, 0 < \varphi < 2\pi \right\} \end{aligned} \quad (2.17)$$

oraz

$$\begin{aligned} \boldsymbol{\psi}_{\mathbf{V}}(r, \varphi, \theta) &= (r \sin \theta \cos \varphi, r \sin \theta \sin \varphi, r \cos \theta)^T = \mathbf{T}, \\ \mathbf{V} &= \left\{ (r, \varphi, \theta)^T \in \mathbb{R}^3 \mid r > 0, 0 < \varphi < 2\pi, 0 < \theta < \pi \right\}. \end{aligned} \quad (2.18)$$

W grupie obrotów  $\text{SO}(3)$  stosowany jest układ współrzędnych typu *oś-kąt* przypisujący każdemu elementowi tej grupy obrót wokół pewnej osi o pewien

<sup>||</sup>Zauważmy, że oba odwzorowania  $\boldsymbol{\psi}_{\mathbf{V}}$  są określone na pewnych nadzbiorach  $\mathbf{V}$ , ale są różnowartościowe tylko na  $\mathbf{V}$ .

kąt. Ma on postać

$$\boldsymbol{\varphi}_{\mathbf{U}}(\mathbf{R}) = (\mathbf{l}(\mathbf{r}), \varphi)^T, \quad (2.19)$$

gdzie  $\mathbf{l}(\mathbf{r})$  jest *osią obrotu* wyznaczoną przez wektor jednostkowy  $\mathbf{r}$  spełniający zależność

$$[\mathbf{r}] = \frac{1}{2 \sin \varphi} (\mathbf{R} - \mathbf{R}^T), \quad \varphi = \arccos \frac{\text{tr} \mathbf{R} - 1}{2}, \quad (2.20)$$

a  $\varphi$  jest kątem obrotu. Dziedziną układu współrzędnych typu oś-kąt jest zbiór

$$\mathbf{U} = \{\mathbf{R} \in \mathbb{SO}(3) \mid -1 < \text{tr} \mathbf{R} < 3\}. \quad (2.21)$$

Zbliżony pod względem formalnym do układu współrzędnych typu oś-kąt jest układ współrzędnych *wykładniczych*

$$\boldsymbol{\varphi}_{\mathbf{U}}(\mathbf{R}) = \mathbf{r}, \quad (2.22)$$

gdzie  $[\mathbf{r}] = \frac{\varphi}{2 \sin \varphi} (\mathbf{R} - \mathbf{R}^T)$ , a kąt  $\varphi$  jest określony jak we wzorze (2.20), którego dziedziną, podobnie jak w przypadku układu współrzędnych typu oś-kąt, jest zbiór (2.21).

Typowym przykładem parametryzacji grupy obrotów jest *parametryzacja Cayleya*

$$\boldsymbol{\psi}_{\mathbf{V}}(\mathbf{r}) = (\mathbb{I}_3 - [\mathbf{r}])^{-1}(\mathbb{I}_3 + [\mathbf{r}]), \quad (2.23)$$

dla której

$$\mathbf{V} = \{\mathbf{r} \in \mathbb{R}^3 \mid \|\mathbf{r}\| < 2\}.$$

Często używane parametryzacje grupy obrotów uzyskuje się przez złożenie trzech kolejnych obrotów wokół osi, parami prostopadłych. Do tego rodzaju parametryzacji należą *kąty Eulera Z-Y-Z*

$$\boldsymbol{\psi}_{\mathbf{V}}(\varphi, \theta, \psi) = \mathbf{R}(Z, \varphi)\mathbf{R}(Y, \theta)\mathbf{R}(Z, \psi), \quad (2.24)$$

których dziedziną jest

$$\mathbf{V} = \{(\varphi, \theta, \psi) \mid 0 < \theta < \pi, 0 < \varphi < 2\pi, 0 < \psi < 2\pi\},$$

oraz parametryzacja przy pomocy kątów zwanych *kołysanie-kiwanie-mysz-kowanie* (Roll-Pitch-Yaw)\*\*

$$\boldsymbol{\psi}_{\mathbf{V}}(\varphi, \theta, \psi) = \mathbf{R}(Z, \varphi)\mathbf{R}(Y, \theta)\mathbf{R}(X, \psi), \quad (2.25)$$

określona na zbiorze

$$\mathbf{V} = \left\{ (\varphi, \theta, \psi) \mid -\frac{\pi}{2} < \theta < \frac{\pi}{2}, 0 < \varphi < 2\pi, 0 < \psi < 2\pi \right\}.$$

We wzorach (2.24), (2.25) symbol  $\mathbf{R}(\text{oś}, \text{kąt})$  oznacza obrót wokół jednej z osi układu ciała o zadany kąt. Łatwo pokazać, że obroty względem osi  $X, Y, Z$  ustalonego układu współrzędnych są zdefiniowane przy pomocy następujących macierzy obrotu:

$$\begin{aligned} \mathbf{R}(X, \alpha) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix}, & \mathbf{R}(Y, \beta) &= \begin{bmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix}, \\ \mathbf{R}(Z, \gamma) &= \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}. \end{aligned} \quad (2.26)$$

Na zakończenie omówimy układ współrzędnych *wykładniczych* w specjalnej grupie euklidesowej  $\mathbb{SE}(3)$ , zdefiniowany formułą

$$\boldsymbol{\varphi}_{\mathbf{U}}(\mathbf{R}, \mathbf{T}) = (\mathbf{r}, \mathbf{t}), \quad (2.27)$$

w której  $[\mathbf{r}] = \frac{\varphi}{2 \sin \varphi} (\mathbf{R} - \mathbf{R}^T)$ ,  $\mathbf{t} = \left( \mathbb{I}_3 - \frac{1}{2}[\mathbf{r}] + \frac{2 \sin \varphi - \varphi(1 + \cos \varphi)}{2\varphi^2 \sin \varphi} [\mathbf{r}]^2 \right) \mathbf{T}$  oraz  $\varphi = \arccos \frac{\text{tr} \mathbf{R} - 1}{2}$ . Dziedziną układu współrzędnych wykładniczych jest

$$\mathbf{U} = \{(\mathbf{R}, \mathbf{T}) \in \mathbb{SE}(3) \mid -1 < \text{tr} \mathbf{R} < 3\}.$$

Układ współrzędnych wykładniczych stanowi odwzorowanie odwrotne do *parametryzacji wykładniczej* grupy  $\mathbb{SE}(3)$ , określonej wyrażeniem

$$\boldsymbol{\psi}_{\mathbf{V}}(\mathbf{r}, \mathbf{t}) = \exp \begin{bmatrix} [\mathbf{r}] & \mathbf{t} \\ \mathbf{0} & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ \mathbf{0} & 1 \end{bmatrix}, \quad (2.28)$$

gdzie  $\exp \mathbf{A} = \sum_{k=0}^{\infty} \frac{\mathbf{A}^k}{k!}$ . Nietrudno zauważyć, że wszystkie układy współrzędnych i parametryzacje (2.17)–(2.28) są lokalnie, a nie globalnie, wzajemnie jednoznaczne (odwracalne).

\*\*Tak wg *Słownika żeglarskiego angielsko-polskiego* pod redakcją W. Petryńskiego, PWN, Warszawa, 1996. Wydaje się, że zaproponowane terminy są bardziej komunikatywne od używanych dotąd w polskich przekładach; wystarczy wyobrazić sobie, że płyniemy łódką w kierunku osi  $X$  układu współrzędnych, którego oś  $Z$  jest skierowana w górę.

## 2.2 Kinematyka układu robotycznego

Niech będzie dany pewien *układ robotyczny* opisany *współrzędnymi uogólnionymi*  $\mathbf{q} = (q_1, q_2, \dots, q_N)^T$  należącymi do pewnego *uniwersum konfiguracyjnego*  $\mathbb{R}^N$ . Załóżmy, że ruch układu jest opisany gładkim przekształceniem  $t \mapsto \mathbf{q}(t)$ . Oznaczmy przez  $\dot{\mathbf{q}} \in \mathbb{R}^N$  *prędkości uogólnione* układu. Przestrzeń  $\mathbb{R}^N \times \mathbb{R}^N \cong \mathbb{R}^{2N}$  położeń i prędkości uogólnionych będziemy nazywać *uniwersum fazowym* układu robotycznego\*.

Wzajemne związki między elementami układu, a także między układem a jego otoczeniem, będziemy reprezentować w formie *ograniczeń (więzów) konfiguracyjnych*

$$\mathbf{f}(\mathbf{q}) = (f_1(\mathbf{q}), f_2(\mathbf{q}), \dots, f_k(\mathbf{q}))^T = \mathbf{0} \quad (2.29)$$

oraz *ograniczeń (więzów) fazowych*

$$\mathbf{A}(\mathbf{q})\dot{\mathbf{q}} = \mathbf{0}. \quad (2.30)$$

Zakładamy, że liczba więzów konfiguracyjnych  $k \leq N$ , a funkcje  $f_1, f_2, \dots, f_k$  są gładkie i niezależne, tzn.

$$\mathbf{f}(\mathbf{q}) = \mathbf{0} \implies \text{rank} \frac{\partial \mathbf{f}}{\partial \mathbf{q}}(\mathbf{q}) = k,$$

oraz że macierz  $\mathbf{A}(\mathbf{q})$  ma rozmiar  $l \times N$ ,  $l \leq N$ , składa się z gładkich funkcji  $a_{ij}(\mathbf{q})$  i jest pełnego rzędu,

$$\text{rank} \mathbf{A}(\mathbf{q}) = l.$$

Postać (2.30) ograniczeń fazowych nazywa się *postacią Pfaffa*. Załóżmy na moment, że ograniczenia fazowe (2.30) nie występują ( $l = 0$ ). Wówczas  $k$  niezależnych ograniczeń konfiguracyjnych wyznacza *rozmaitość konfiguracyjną*

$$\mathcal{Q} = \{ \mathbf{q} \in \mathbb{R}^N \mid f_1(\mathbf{q}) = \dots = f_k(\mathbf{q}) = 0 \} \quad (2.31)$$

układu, której wymiar  $\dim \mathcal{Q} = N - k = n$ , z przestrzenią styczną  $\mathbf{T}_{\mathbf{q}}\mathcal{Q} = \text{Ker} \frac{\partial \mathbf{f}}{\partial \mathbf{q}}(\mathbf{q})$ . Liczbę  $n$  nazywamy liczbą *stopni swobody* układu robotycznego. W rezultacie, ruch układu jest ograniczony do rozmaitości  $\mathcal{Q} \subset \mathbb{R}^N$ .

---

\* Jeżeli ruch układu nie podlega ograniczeniom, uniwersum fazowe pokrywa się z przestrzenią fazową (przestrzenią stanu) układu dynamicznego zdefiniowanego równaniami Eulera-Lagrange'a.



Obecność ograniczeń fazowych (2.30) może, ale nie musi, prowadzić do dalszego ograniczenia dopuszczalnych konfiguracji układu. Zależy to od własności ograniczeń fazowych zwanej *holonomicznością*. Ograniczenia fazowe nazywamy holonomicznymi, jeżeli jest możliwe ich scałkowanie i zastąpienie ograniczeniami postaci (2.29). Zatem, w przypadku ograniczeń holonomicznych istnieje odwzorowanie  $\mathbf{h}(\mathbf{q}) = (h_1(\mathbf{q}), h_2(\mathbf{q}), \dots, h_l(\mathbf{q}))^T$ , takie że (2.30) jest równoważne warunkowi

$$\mathbf{h}(\mathbf{q}) = \mathbf{0}.$$

Łatwo zauważyć, że holonomiczne ograniczenia fazowe można dołączyć do ograniczeń konfiguracyjnych. Inaczej jest w przypadku ograniczeń nieholonomicznych. Nieholonomiczność ograniczeń (2.30) oznacza, że scałkowanie systemu (2.30) nie jest możliwe, a obecność tych ograniczeń nie zmniejsza osiągalności konfiguracji w obrębie rozmaitości konfiguracyjnej. Utrudnieniu może ulec natomiast sposób osiągania pewnych konfiguracji. Aby przeanalizować bardziej szczegółowo znaczenie nieholonomiczności ograniczeń fazowych założymy, że układ nie podlega ograniczeniom konfiguracyjnym ( $k = 0$ ,  $n = N$ ). Wówczas z (2.30) wynika, że w każdej konfiguracji  $\mathbf{q} \in \mathbb{R}^n$  dopuszczalne prędkości układu muszą należeć do przestrzeni zerowej (jądra) macierzy  $\mathbf{A}(\mathbf{q})$ ,

$$\dot{\mathbf{q}} \in \text{Ker } \mathbf{A}(\mathbf{q}). \quad (2.32)$$

Niech  $\mathbf{G}(\mathbf{q}) = [g_1(\mathbf{q}), g_2(\mathbf{q}), \dots, g_{n-l}(\mathbf{q})]$  będzie macierzą, której kolumnami są wektory rozpinające przestrzeń liniową  $\text{Ker } \mathbf{A}(\mathbf{q})$  w konfiguracji  $\mathbf{q}$ . Z niezależności ograniczeń fazowych wynika, że w każdym punkcie rząd  $\mathbf{G}(\mathbf{q})$  jest pełny,

$$\text{rank } \mathbf{G}(\mathbf{q}) = n - l = m,$$

natomiast własność (2.32) jest równoważna istnieniu wektora  $\mathbf{u} \in \mathbb{R}^m$ , takiego że

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{q})u_i. \quad (2.33)$$

Pola wektorowe  $\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_m$  wyznaczają w uniwersum fazowym obiekt geometryczny  $\mathcal{G} = \text{span}_{\mathbb{C}^\infty}\{\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_m\}$ , zwany *dystrybucją*. Elementami dystrybucji są kombinacje pól  $\mathbf{g}_1, \dots, \mathbf{g}_m$  mnożonych przez funkcje gładkie<sup>†</sup>. Spełnienie ograniczeń fazowych oznacza, że w każdym punkcie uniwersum konfiguracyjnego prędkość układu należy do dystrybucji  $\mathcal{G}$ , a więc

<sup>†</sup>Formalnie, pola wektorowe należące do dystrybucji  $\mathcal{G}$  mają postać  $\sum_{i=1}^m \alpha_i \mathbf{g}_i$ , gdzie  $\alpha_i \in \mathbb{C}^\infty(\mathbb{R}^n)$ .

jest rozpięta przez kolumny macierzy  $\mathbf{G}$ . Mając wyznaczoną dystrybucję  $\mathcal{G}$ , możemy się posługiwać następującym kryterium nieholonomiczności ograniczeń.

**Twierdzenie 2.2.1** *Niech  $\mathcal{G} = \text{span}_{\mathbb{C}^\infty}\{\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_m\}$ . Zdefiniujmy ciąg dystrybucji*

$$\mathcal{G}^0 = \mathcal{G}, \quad \mathcal{G}^1 = \mathcal{G}^0 + [\mathcal{G}^0, \mathcal{G}^0], \dots, \quad \mathcal{G}^j = \mathcal{G}^{j-1} + [\mathcal{G}^{j-1}, \mathcal{G}^0], \dots,$$

$j = 1, 2, \dots$ , w którym operacja  $[\ , \ ]$  oznacza nawias Liego pól wektorowych i jest zdefiniowana formułą

$$[\mathbf{X}, \mathbf{Y}](\mathbf{q}) = \frac{\partial \mathbf{Y}}{\partial \mathbf{q}} \mathbf{X}(\mathbf{q}) - \frac{\partial \mathbf{X}}{\partial \mathbf{q}} \mathbf{Y}(\mathbf{q}). \quad (2.34)$$

Wówczas, jeżeli dla pewnego  $j = r$  zachodzi  $\dim \mathcal{G}^r(\mathbf{q}) = n$ , to ograniczenia (2.30) są nieholonomiczne. Liczba  $r$  nazywa się stopniem nieholonomiczności dystrybucji  $\mathcal{G}$ .

Ciąg  $\mathcal{G}^0 \subset \mathcal{G}^1 \subset \dots \subset \mathcal{G}^j \subset \dots$  nazywamy *filtracją* dystrybucji  $\mathcal{G}$ . Filtracja jest regularna, jeżeli każda dystrybucja  $\mathcal{G}^j$  ma stały wymiar, to znaczy w każdym punkcie  $\mathbf{q}$   $\dim \mathcal{G}^j(\mathbf{q}) = r_j = \text{const}$ . W przypadku filtracji regularnej ciąg wymiarów  $r_0 \leq r_1 \leq \dots \leq r_k \leq \dots$  ustala się w skończonej liczbie kroków  $p \leq n$  na pewnym poziomie  $r_p$ . Ciąg  $(r_0, r_1, \dots, r_p)$  nazywamy *wektorem wzrostu* dystrybucji  $\mathcal{G}$ . Jeżeli  $r_p < n$ , to ograniczenia fazowe (2.30) są holonomiczne. W szczególności, jeżeli  $l = n - 1$  (tzn.  $p = 0$  i  $r_p = 1$ ), to dystrybucja  $\mathcal{G}$  jest generowana przez jedno pole wektorowe i ograniczenia (2.30) są holonomiczne.

Jeżeli występują ograniczenia konfiguracyjne (2.29), a zatem ruch układu jest ograniczony do rozmaitości konfiguracyjnej  $\mathcal{Q}$ , nałożenie ograniczeń nieholonomicznych (2.30) powinno być zgodne z istniejącymi ograniczeniami konfiguracyjnymi. Rozumiemy przez to spełnienie następującego warunku

$$\text{Ker } \mathbf{A}(\mathbf{q}) \subset \text{Ker } \frac{\partial \mathbf{f}}{\partial \mathbf{q}}(\mathbf{q}),$$

co oznacza, że w każdym punkcie  $\mathbf{q} \in \mathcal{Q}$  dystrybucja  $\mathcal{G}$  spełnia warunek  $\mathcal{G}(\mathbf{q}) \subset \mathbf{T}_{\mathbf{q}}\mathcal{Q}$ , skąd wynika, że dopuszczalne prędkości układu należą do podprzestrzeni  $\mathcal{G}(\mathbf{q})$  przestrzeni stycznej do  $\mathcal{Q}$ . Warunek nieholonomiczności przybierze wtedy postać  $\dim \mathcal{G}^p(\mathbf{q}) = r_p = \dim \mathcal{Q}^\ddagger$ .

<sup>‡</sup>Przykłady ograniczeń konfiguracyjnych i fazowych związanych z zadaniem toczenia się koła bez poślizgu oraz z kinematyką spadającego kota podajemy w rozdziale 8.

Zadanie, jakie ma wykonać układ robotyczny polega na celowym oddziaływaniu (akcji) na otoczenie układu. Opis formalny tego oddziaływania na poziomie uniwersum fazowego wymaga wprowadzenia obiektu zwanego *rozmaitością zadaniową*. W przypadku pojedynczego manipulatora oddziaływanie na otoczenie zachodzi za pośrednictwem efektora, a rozmaitością zadaniową jest zbiór położeń i orientacji efektora, czyli grupa  $\mathbb{SE}(3)$  (lub pewna podgrupa  $\mathbb{SE}(3)$ ), dla jednokołowego robota mobilnego poruszającego się po płaszczyźnie, którego oddziaływanie na otoczenie zachodzi za pośrednictwem koła, rozmaitością zadaniową będzie grupa  $\mathbb{SE}(2) \cong \mathbb{R}^2 \times \mathbb{S}^1$ , złożona z położeń i orientacji ciała sztywnego w  $\mathbb{R}^2$ . Dla ustalonej rozmaitości zadaniowej, możliwości wywierania przez układ określonych akcji na otoczenie opisujemy przy pomocy odwzorowania zwanego *kinematyką układu robotycznego*. W przypadku układu holonomicznego (np. manipulatora) kinematyka jest pewnym odwzorowaniem

$$\mathbf{K} : \mathcal{Q} \longrightarrow \mathcal{Z}, \quad \mathbf{z} = \mathbf{K}(\mathbf{q}) \quad (2.35)$$

rozmaitości konfiguracyjnej układu w rozmaitość zadaniową. Kinematyka  $\mathbf{K}$  układu nie zawsze jest zadana w sposób jawny. Na przykład, w przypadku układów zawierających zamknięte łańcuchy kinematyczne, kinematyka może być wyrażona w formie uwikłanej za pośrednictwem ograniczeń konfiguracyjnych  $\mathbf{f}(\mathbf{q}, \mathbf{z}) = \mathbf{0}$ . Jeżeli w układzie istnieją nieholonomiczne ograniczenia fazowe, do opisu kinematyki służy *układ sterowania* postaci

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u}, \quad (2.36)$$

w którym wektor  $\mathbf{u}$  należy do pewnej przestrzeni sterowań.

Aby zrozumieć znaczenie sterowań  $\mathbf{u}$  w układzie nieholonomicznym, przeanalizujemy teraz dokładniej formułę (2.30) opisującą ograniczenia fazowe. W pewnym otoczeniu punktu  $\mathbf{q}^{\S}$  formułę tę można przepisać w postaci

$$[\mathbf{A}_1(\mathbf{q}) \quad \mathbf{A}_2(\mathbf{q})] \dot{\mathbf{q}} = \mathbf{0}, \quad (2.37)$$

w której macierz  $\mathbf{A}_2(\mathbf{q})$  jest rozmiaru  $l \times l$  i pełnego rzędu. Oczywiście, ograniczenia (2.37) będą spełnione<sup>¶</sup>, jeżeli pomnożymy obie strony równości (2.37) lewostronnie przez macierz odwrotną  $\mathbf{A}_2^{-1}(\mathbf{q})$ . W rezultacie otrzymamy następującą lokalnie równoważną postać ograniczeń fazowych

$$[\mathbf{A}_2^{-1}(\mathbf{q})\mathbf{A}_1(\mathbf{q}) \quad \mathbb{I}_l] \dot{\mathbf{q}} = [-\mathbf{W}(\mathbf{q}) \quad \mathbb{I}_l] \dot{\mathbf{q}} = \mathbf{0}$$

<sup>§</sup>W razie potrzeby, po przenieumerowaniu współrzędnych uogólnionych.

<sup>¶</sup>W każdym punkcie, w którym macierz  $\mathbf{A}_2(\mathbf{q})$  jest nieosobliwa.

prowadzącą do macierzy  $\mathbf{G}(\mathbf{q}) = \begin{bmatrix} \mathbb{I}_{N-l} \\ \mathbf{W}(\mathbf{q}) \end{bmatrix}$  układu sterowania (2.36). Podzielmy współrzędne uogólnione  $\mathbf{q}$  na dwie grupy  $\mathbf{q} = (\mathbf{q}^{1T}, \mathbf{q}^{2T})^T$ , gdzie  $\dim \mathbf{q}^1 = N - l = m$ ,  $\dim \mathbf{q}^2 = l$ . Przy takim podziale kinematyka układu (2.36) przyjmie postać

$$\dot{\mathbf{q}}^1 = \mathbf{u}, \quad \dot{\mathbf{q}}^2 = \mathbf{W}(\mathbf{q})\mathbf{u}. \quad (2.38)$$

Ze wzoru (2.38) wynika, że sterowania  $\mathbf{u}$  występujące w równaniach kinematyki układu nieholonomicznego mogą być interpretowane jako pewne składowe prędkości uogólnionych układu.

## 2.3 Kinematyka manipulatora

*Manipulator* o  $n$  stopniach swobody jest układem robotycznym złożonym z  $n$  ciał sztywnych zwanych *ramionami*, połączonych za pośrednictwem *przegubów*. Jeden koniec łańcucha ramion tworzącego manipulator jest związany z nieruchomą bazą (np. podłożem), drugi koniec pozostaje swobodny. Parę złożoną z przegubu i następującego po nim ramienia będziemy nazywać *ogniwem* manipulatora. Dwa ramiona połączone przegubem noszą nazwę *pary kinematycznej*. W układach robotycznych najczęściej są spotykane pary kinematyczne V klasy, to znaczy takie, które posiadają jeden stopień swobody ruchu w przegubie. Istnieją dwa rodzaje przegubów spełniających ten warunek: *przeguby obrotowe* i *przeguby przesuwne*. Geometrycznie, przegub obrotowy można zdefiniować zadając punkt w przestrzeni (środek przegubu) i wektor w tym punkcie (oś obrotu przegubu). Umieścmy układ współrzędnych związany z pierwszym elementem pary obrotowej w środku przegubu w taki sposób, by oś  $Z$  pokrywała się z osią przegubu. Niech układ związany z drugim elementem pary obrotowej znajduje się również w środku przegubu i niech oś  $Z$  tego układu będzie osią przegubu. Rozważmy ruch jednego elementu pary względem drugiego. Jak wiadomo, rozmaitość konfiguracyjna ciała sztywnego,  $\mathbb{SE}(3)$ , jest sześciowymiarowa i składa się z przesunięć  $\mathbf{T} \in \mathbb{R}^3$  oraz obrotów  $\mathbf{R} \in \mathbb{SO}(3)$ . Ustalenie położenia środka przegubu wyznacza wektor  $\mathbf{T}$  ( $\mathbf{T} = \mathbf{0}$ ). Co więcej, ustalenie osi obrotu wymaga, by  $\mathbf{R}\mathbf{e}_3 = \mathbf{e}_3$ , czyli, biorąc pod uwagę ortogonalność macierzy obrotu,  $\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & 0 \\ r_{21} & r_{22} & 0 \\ 0 & 0 & 1 \end{bmatrix}$ . Z ortogonalności macierzy  $\mathbf{R}$  wynika również, że  $r_{11}^2 + r_{12}^2 = 1$ ,  $r_{21}^2 + r_{22}^2 = 1$ ,  $r_{11}r_{21} + r_{12}r_{22} = 0$ . Te trzy ograniczenia będą spełnione, jeżeli  $r_{11} = \cos \varphi$ ,  $r_{12} = -\sin \varphi$ ,  $r_{21} = \sin \varphi$ ,  $r_{22} = \cos \varphi$ . Otrzymana w ten sposób macierz  $\mathbf{R} = \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}$  reprezentuje obrót

wokół osi  $Z$  o kąt  $\varphi$ , skąd wynika, że rozmierność konfiguracyjna pary kinematycznej  $V$  klasy wykonującej ruch w zamocowanym, nieograniczonym przegubie obrotowym ma postać okręgu jednostkowego  $\mathbb{S}^1$ . Ograniczenie zakresu ruchu w przegubie może wymagać zastąpienia okręgu pewnym łukiem okręgu. W przypadku, gdy ruch ciała sztywnego zachodzi względem zamocowanego, nieograniczonego przegubu przesuwne, rozumowanie analogiczne do powyższego prowadzi do wniosku, że rozmierność konfiguracyjna może być utożsamiona ze zbiorem liczb rzeczywistych  $\mathbb{R}$  lub z pewnym przedziałem liczb rzeczywistych, jeżeli zakres ruchu w przegubie jest ograniczony.

Biorąc pod uwagę powyższe spostrzeżenia, nietrudno dojść do wniosku, że dla manipulatora o  $n$  stopniach swobody, posiadającego  $r$  przegubów obrotowych i  $n-r$  przegubów przesuwnych (wszystkie nieograniczone), rozmierność konfiguracyjna może być zdefiniowana jako iloczyn kartezjański torusa  $\mathbb{T}^r$  i przestrzeni  $\mathbb{R}^{n-r}$

$$Q = \underbrace{\mathbb{S}^1 \times \mathbb{S}^1 \times \cdots \times \mathbb{S}^1}_{r \text{ razy}} \times \underbrace{\mathbb{R}^1 \times \mathbb{R}^1 \times \cdots \times \mathbb{R}^1}_{(n-r) \text{ razy}} \cong \mathbb{T}^r \times \mathbb{R}^{n-r}. \quad (2.39)$$

Rozmierność konfiguracyjną (2.39) będziemy odtąd nazywać *rozmiernością przegubową*. Naturalną parametryzację rozmierności przegubowej tworzy zbiór kątów obrotu w przegubach obrotowych  $q_1, q_2, \dots, q_r \in (0, 2\pi)$  oraz zestaw przesunięć w przegubach przesuwnych  $q_{r+1}, q_{r+2}, \dots, q_n \in \mathbb{R}$ .

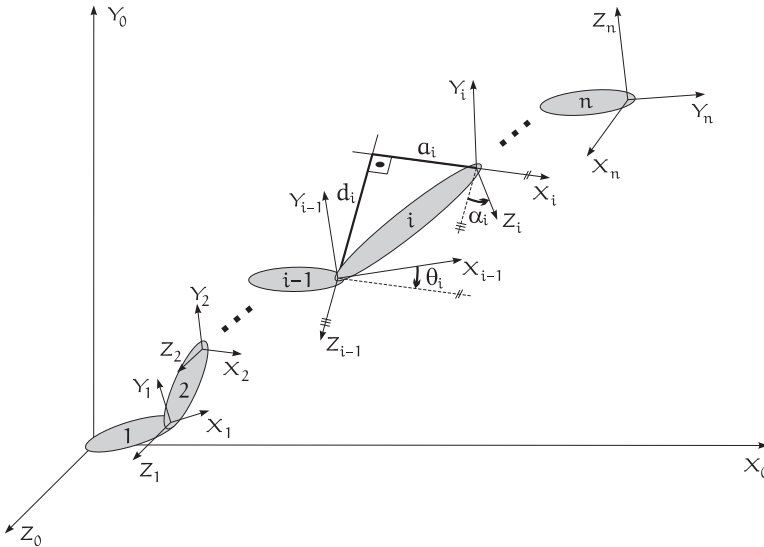
Ostatnie ramię manipulatora, licząc od bazy, służy do oddziaływania na otoczenie i dlatego jest nazywane *efektorem* manipulatora. Jak zauważyliśmy w poprzednim podrozdziale, zadanie jakie ma wykonać manipulator formułuje się zwykle przez określenie pożądanego położenia i orientacji efektora. W związku z tym, będziemy zakładać, że rozmierność zadaniowa manipulatora zawiera się w specjalnej grupie euklidesowej,  $\mathcal{Z} \subset \mathbb{SE}(3)$ , a przez *kinematykę manipulatora* będziemy rozumieć gładkie (analityczne) odwzorowanie

$$\mathbf{K}: Q \longrightarrow \mathcal{Z} \subset \mathbb{SE}(3) \quad (2.40)$$

rozmierności przegubowej w rozmierność zadaniową.

### 2.3.1 Reprezentacja Denavita-Hartenberga

Odwzorowanie  $\mathbf{K}$  określa położenie i orientację efektora w funkcji położen przyjmowanych przez przeguby manipulatora. Powszechnie przyjęty w ro-



Rysunek 2.3 Schemat kinematyki manipulatora.

botyce sposób reprezentowania kinematyki manipulatora pochodzi od Denavita i Hartenberga. Algorytm Denavita-Hartenberga wyznaczania kinematyki polega na związaniu z każdym przegubem manipulatora lokalnego układu współrzędnych, a następnie określeniu ciągu transformacji sąsiednich układów współrzędnych, i prowadzi do wyliczenia kinematyki manipulatora jako złożenia tych transformacji. Niech będzie dany manipulator o  $n$  przegubach, obrotowych bądź przesuwanych, i układ przestrzeni  $X_0Y_0Z_0$  (podstawowy układ współrzędnych). Zakładamy, że przegub nr 1 został umieszczony w początku układu przestrzeni w taki sposób, że oś  $Z$  układu przestrzeni pokrywa się z osią tego przegubu. Początek układu współrzędnych  $X_{i-1}Y_{i-1}Z_{i-1}$  związanego z ramieniem nr  $i-1$  manipulatora leży na osi  $i$ -tego przegubu,  $i = 2, 3, \dots, n$ . Ruch  $i$ -tego przegubu zachodzi względem osi  $Z$  układu  $(i-1)$ -szego. Układ  $X_nY_nZ_n$  związany z efekтором umieszczamy tak, aby jego oś  $X$  przecinała się z osią  $Z_{n-1}$  i była do niej prostopadła. Rozmieszczenie układów współrzędnych ilustruje rysunek 2.3. Załóżmy, że wyznaczyliśmy położenie i orientację układu  $X_{i-1}Y_{i-1}Z_{i-1}$ . Położenie i orientację układu  $X_iY_iZ_i$  definiujemy na podstawie znajomości położenia i orientacji układu  $(i-1)$ -szego. W tym celu prowadzimy normalną do osi przegubów nr  $i, i+1$ . Wyznaczamy punkty przecięcia normalnej z osiami. Następnie obracamy układ  $(i-1)$ -szy wokół osi  $i$ -tego przegubu, tak by oś  $X$  układu stała się równoległa do normalnej, przesuwamy ten układ wzdłuż osi

$i$ -tego przegubu aż do punktu przecięcia tej osi z normalną, z kolei przesuwamy układ wzdłuż normalnej, do punktu przecięcia z osią przegubu nr  $i + 1$ , i w końcu obracamy układ wokół osi  $\times$  aż do pokrycia się osi  $Z$  układu z osią przegubu  $(i + 1)$ -szego\*. Tę procedurę powtarzamy dla kolejnych ogniw dochodząc do układu związanego z efekтором. Z formalnego punktu widzenia, transformacja układu numer  $i - 1$  w układ o numerze  $i$  może być opisana przy pomocy złożenia

$$\mathbf{A}_{i-1}^i(q_i) = \mathbf{Rot}(Z, \theta_i) \mathbf{Trans}(Z, d_i) \mathbf{Trans}(\times, a_i) \mathbf{Rot}(\times, \alpha_i) \quad (2.41)$$

*elementarnych obrotów i przesunięć*

$$\mathbf{Rot}(\text{oś}, \text{kąt}) = \begin{bmatrix} \mathbf{R}(\text{oś}, \text{kąt}) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix},$$

$$\mathbf{Trans}(\text{oś}, \text{przesunięcie}) = \begin{bmatrix} \mathbb{I}_3 & \text{przesunięcie} \cdot \text{oś} \\ \mathbf{0} & 1 \end{bmatrix}.$$

Dla przegubu obrotowego zmienna przegubowa  $q_i = \theta_i$ , zaś dla przegubu przesuwnego  $q_i = d_i$ . Parametry występujące w zależności (2.41) nazywamy *parametrami Denavita-Hartenberga* manipulatora. Parametry, które nie są zmiennymi przegubowymi charakteryzują geometrię ogniwa manipulatora i noszą nazwę *parametrów geometrycznych*. *Kinematyka manipulatora* określa położenie i orientację układu efektora względem układu podstawowego, i jest opisana złożeniem transformacji (2.41)

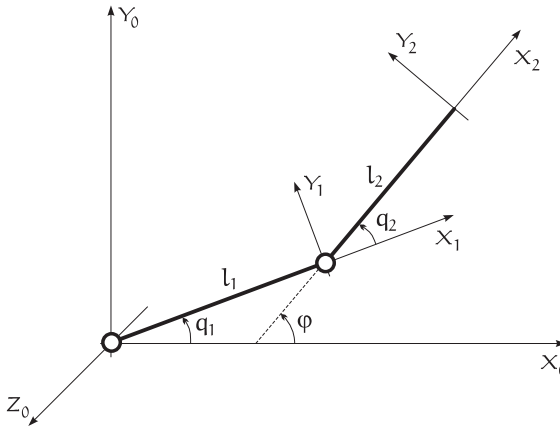
$$\mathbf{K}(\mathbf{q}) = \prod_{i=1}^n \mathbf{A}_{i-1}^i(q_i) = \begin{bmatrix} \mathbf{R}_0^n(\mathbf{q}) & \mathbf{T}_0^n(\mathbf{q}) \\ \mathbf{0} & 1 \end{bmatrix}. \quad (2.42)$$

Odwzorowanie (2.42) nazywamy *reprezentacją kinematyki manipulatora wg Denavita-Hartenberga*.

Poniżej wyprowadzimy modele matematyczne kinematyki kilku przykładowych manipulatorów. Pozwoli to Czytelnikowi na prześledzenie sposobu posługiwania się algorytmem Denavita-Hartenberga. Wyprowadzone modele zostaną wykorzystane w rozdziale 3 przy omawianiu algorytmów kinematyki odwrotnej.

---

\* Jeżeli normalna nie jest wyznaczona jednoznacznie, początek układu  $X_i Y_i Z_i$  umieszczamy w środku przegubu  $(i+1)$ -szego. Jeżeli osie  $Z_{i-1}$ ,  $Z_i$  przecinają się, początek układu  $X_i Y_i Z_i$  umieszczamy w punkcie przecięcia osi, a oś  $X_i$  orientujemy w kierunku  $Z_{i-1} \times Z_i$ .



Rysunek 2.4 Manipulator typu podwójne wahadło.

**Przykład 2.3.1 (Manipulator typu podwójne wahadło)**

Bardzo prostą konstrukcją manipulatora jest przedstawiony na rysunku 2.4 manipulator typu podwójne wahadło, zbudowany z dwóch ramion i dwóch przegubów obrotowych. Osie obrotu przegubów manipulatora są prostopadłe do płaszczyzny  $X_0Y_0$  (równoległe do osi  $Z_0$ ). Przy wyborze tego manipulatora jako pierwszego przykładu kierowaliśmy się nie tylko przesłankami natury ilustracyjnej, lecz także praktycznej: strukturę kinematyczną podwójnego wahadła posiada manipulator eksperymentalny *EDDA*, którego model wykorzystujemy do badania algorytmów sterowania, a także łańcuch kinematyczny utworzony przez pierwsze dwa ogniwa manipulatora przemysłowego typu *SCARA*. Przyjmijmy, że ruch w obu przegubach manipulatora jest nieograniczony, a długości ramion wynoszą  $l_1$  i  $l_2$ .

Rozpatrywany manipulator jest manipulatorem klasy  $2R^\dagger$ , a zatem jego rozmaitość konfiguracyjna  $\mathcal{Q}$  może być utożsamiona z dwuwymiarowym torusem  $\mathbb{T}^2$ , którego punkty są reprezentowane przez kąty obrotu w pierwszym i drugim przegubie,  $\mathbf{q} = (q_1, q_2)^\top$ . Zadanie kinematyki manipulatora będzie polegało na wyrażeniu położenia i orientacji efektora manipulatora w podstawowym układzie współrzędnych  $X_0Y_0Z_0$  jako funkcji konfiguracji manipulatora  $\mathbf{q}$ , czyli na wyznaczeniu odwzorowania

$$\mathbf{K} : \mathbb{T}^2 \longrightarrow \text{SE}(3). \quad (2.43)$$

Zgodnie z algorytmem Denavita-Hartenberga, zaczynamy od zdefiniowania podstawowego układu współrzędnych (układu przestrzeni) oraz roz-

<sup>†</sup>Co oznacza, że posiada dwa przeguby typu obrotowego (R).



ogniwo	$\theta_i$	$d_i$	$a_i$	$\alpha_i$
1	$q_1$	0	$l_1$	0
2	$q_2$	0	$l_2$	0

Tabela 2.1 Parametry Denavita-Hartenberga manipulatora typu podwójne wahadło.

mieszczenia lokalnych układów współrzędnych (układów ciał) związanych z ramionami manipulatora. Przy wyborze układu podstawowego  $X_0Y_0Z_0$  musimy jedynie zadbać, by oś  $Z$  tego układu pokrywała się z osią obrotu pierwszego przegubu manipulatora. Lokalny układ współrzędnych  $X_1Y_1Z_1$ <sup>‡</sup> związany z pierwszym ramieniem manipulatora umieszczamy tak, że oś  $Z$  tego układu pokrywa się z osią obrotu przegubu drugiego. Ponieważ w rozpatrywanym przypadku osie obrotu przegubów są do siebie równoległe (istnieje nieskończenie wiele normalnych do nich), początek układu  $X_1Y_1Z_1$  wybieramy w środku przegubu drugiego. Początek następującego po nim układu  $X_2Y_2Z_2$  związanego z efektem manipulatora umieszczamy na końcu drugiego ramienia, a sam układ orientujemy w taki sposób, aby oś  $Z_2$  była równoległa do osi  $Z_1$ .

Po określeniu układów współrzędnych wyznaczamy parametry Denavita-Hartenberga rozpatrywanego manipulatora. Jak już powiedzieliśmy, kąty obrotu  $\theta_1$  i  $\theta_2$  są równe odpowiednio zmiennym  $q_1$  i  $q_2$  parametryzującym różnorodność przegubową. Przesunięcia  $a_1$  i  $a_2$  wzdłuż osi  $X$  lokalnych układów współrzędnych wynoszą odpowiednio  $l_1$  i  $l_2$ . Pozostałe parametry Denavita-Hartenberga, tzn. przesunięcia  $d_1$ ,  $d_2$  razem z kątami  $\alpha_1$  i  $\alpha_2$  są równe zero. W tabeli 2.1 zebrano wszystkie wymienione parametry manipulatora.

Określmy teraz kinematykę manipulatora w postaci (2.43). W tym celu musimy znaleźć macierze transformacji między kolejnymi, lokalnymi układami współrzędnych. Macierz  $\mathbf{A}_0^1(q_1) : X_0Y_0Z_0 \mapsto X_1Y_1Z_1$  jest złożeniem obrotu i przesunięcia

$$\mathbf{A}_0^1(q_1) = \mathbf{Rot}(Z, q_1)\mathbf{Trans}(X, l_1),$$

gdyż kąt skręcenia osi przegubów  $\alpha_1 = 0$  i przesunięcie układu  $d_1 = 0$ . Po wprowadzeniu oznaczeń  $\sin q_i = s_i$ ,  $\cos q_i = c_i$ , na podstawie wzoru (2.41),

<sup>‡</sup>Nie pokazana na rysunku oś  $Z_1$  dopełnia układ  $X_1Y_1$  do układu prawoskrętnego.

macierz  $\mathbf{A}_0^1$  przyjmuje postać

$$\mathbf{A}_0^1(q_1) = \begin{bmatrix} c_1 & -s_1 & 0 & l_1 c_1 \\ s_1 & c_1 & 0 & l_1 s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.44)$$

Analogicznie, macierz  $\mathbf{A}_1^2(q_2) : X_1 Y_1 Z_1 \mapsto X_2 Y_2 Z_2$  można wyznaczyć jako

$$\mathbf{A}_1^2(q_2) = \mathbf{Rot}(Z, q_2) \mathbf{Trans}(X, l_2) = \begin{bmatrix} c_2 & -s_2 & 0 & l_2 c_2 \\ s_2 & c_2 & 0 & l_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.45)$$

Mając wyliczone transformacje  $\mathbf{A}_{i-1}^i(q_i)$  między sąsiednimi układami współrzędnych, definiujemy kinematykę manipulatora jako

$$\mathbf{K}(\mathbf{q}) = \mathbf{A}_0^1(q_1) \mathbf{A}_1^2(q_2) : X_0 Y_0 Z_0 \mapsto X_2 Y_2 Z_2,$$

czyli

$$\mathbf{K}(\mathbf{q}) = \begin{bmatrix} c_{12} & -s_{12} & 0 & l_1 c_1 + l_2 c_{12} \\ s_{12} & c_{12} & 0 & l_1 s_1 + l_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.46)$$

przy oznaczeniach  $s_{12} = \sin(q_1 + q_2)$  i  $c_{12} = \cos(q_1 + q_2)$ . ■

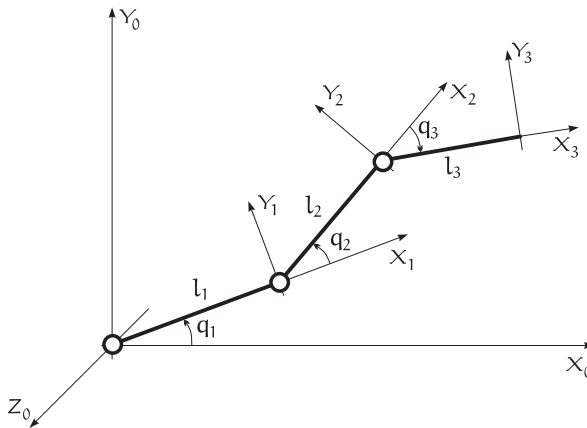
### Przykład 2.3.2 (Manipulator typu potrójne wahadło)

Innym, prostym przykładem manipulatora jest manipulator typu potrójne wahadło, którego łańcuch kinematyczny pokazano schematycznie na rysunku 2.5, a parametry Denavita-Hartenberga zamieszczono w tabeli 2.2. Rozmaitość konfiguracyjna  $\mathcal{Q}$  rozpatrywanego tu manipulatora ma postać trójwymiarowego torusa  $\mathbb{T}^3$ , którego punkty są reprezentowane przez wektor  $\mathbf{q} = (q_1, q_2, q_3)^T$  kątów obrotu w przegubach.

Do określenia kinematyki

$$\mathbf{K} : \mathbb{T}^3 \longrightarrow \text{SE}(3)$$

manipulatora zastosujemy algorytm Denavita-Hartenberga, dzięki czemu



Rysunek 2.5 Manipulator typu potrójne wahadło.

ogniwo	$\theta_i$	$d_i$	$a_i$	$\alpha_i$
1	$q_1$	0	$l_1$	0
2	$q_2$	0	$l_2$	0
3	$q_3$	0	$l_3$	0

Tabela 2.2 Parametry Denavita-Hartenberga manipulatora typu potrójne wahadło.

otrzymamy wyrażenia

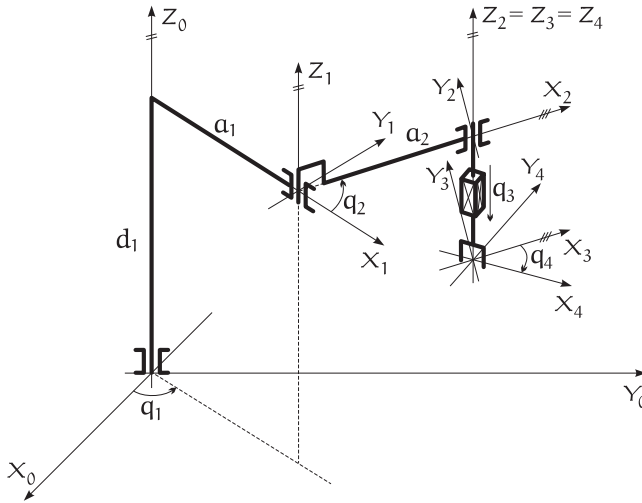
$$\mathbf{K}(\mathbf{q}) = \begin{bmatrix} c_{123} & -s_{123} & 0 & l_1 c_1 + l_2 c_{12} + l_3 c_{123} \\ s_{123} & c_{123} & 0 & l_1 s_1 + l_2 s_{12} + l_3 s_{123} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.47)$$

gdzie przez  $s_{12}$ ,  $c_{12}$  oznaczono, odpowiednio,  $\sin(q_1 + q_2)$  i  $\cos(q_1 + q_2)$ , a przez  $s_{123}$ ,  $c_{123}$  —  $\sin(q_1 + q_2 + q_3)$  i  $\cos(q_1 + q_2 + q_3)$ . ■

### Przykład 2.3.3 (Manipulator typu SCARA)

Manipulator SCARA jest manipulatorem przemysłowym, którego schemat kinematyczny przedstawiono na rysunku 2.6<sup>§</sup>. Widzimy, że oprócz dwóch przegubów obrotowych tworzących łańcuch kinematyczny analogiczny do

<sup>§</sup>Manipulator SCARA znajduje się w Laboratorium Centrum Systemów Produkcyjnych Instytutu Technologii Maszyn i Automatykacji Politechniki Wrocławskiej. Udostępnienie danych tego manipulatora zawdzięczamy uprzejmości dra inż. A. Kocełucha.



Rysunek 2.6 Struktura kinematyczna manipulatora typu *SCARA*.

ogniwo	$\theta_i$	$d_i$	$a_i$	$\alpha_i$
1	$q_1$	$d_1$	$a_1$	0
2	$q_2$	0	$a_2$	0
3	0	$q_3$	0	0
4	$q_4$	0	0	0

Tabela 2.3 Parametry Denavita-Hartenberga manipulatora typu *SCARA*.

podwójnego wahadła, manipulator posiada jeszcze dwa przeguby: przesuwny i obrotowy; ma zatem cztery stopnie swobody (jest manipulatorem klasy RRTR). Cechą charakterystyczną konstrukcji manipulatora typu *SCARA* jest równoległość osi obrotu/przesuwu przegubów. Mając przypisane układy współrzędnych zgodnie z algorytmem Denavita-Hartenberga, po określeniu parametrów manipulatora (zamieszczonych w tabeli 2.3), możemy przystąpić do wyznaczenia kinematyki

$$\mathbf{K}(\mathbf{q}) : \mathcal{Q} \cong \mathbb{T}^3 \times \mathbb{R} \longrightarrow \text{SE}(3),$$

gdzie  $\mathbf{q} = (q_1, q_2, q_3, q_4)^T$ . Przekształcenia  $\mathbf{A}_{i-1}^i(q_i)$ ,  $i = 1, \dots, 4$  między poszczególnymi układami współrzędnych są dane jako

$$\mathbf{A}_0^1(q_1) = \mathbf{Rot}(Z, q_1) \mathbf{Trans}(Z, d_1) \mathbf{Trans}(X, a_1),$$

$$\mathbf{A}_1^2(q_2) = \mathbf{Rot}(Z, q_2) \mathbf{Trans}(X, a_2),$$

ogniwo	$\theta_i$	$d_i$	$a_i$	$\alpha_i$
1	0	$q_1$	0	$\pi/2$
2	$q_2$	$d_1$	0	$\pi/2$
3	$q_3$	0	$a_2$	0
4	$q_4 - q_3$	0	$a_3$	0
5	$q_5 - q_4$	0	0	$\pi/2$
6	$q_6$	$d_6$	0	$-\pi/2$

**Tabela 2.4** Parametry Denavita-Hartenberga manipulatora IRb-6 zamontowanego na torze jezdnym.

$$\mathbf{A}_2^3(q_3) = \mathbf{Trans}(Z, q_3),$$

$$\mathbf{A}_3^4(q_4) = \mathbf{Rot}(Z, q_4).$$

Zgodnie z zależnością (2.42), kinematyka manipulatora jest odwzorowaniem

$$\mathbf{K}(\mathbf{q}) = \mathbf{A}_0^1(q_1)\mathbf{A}_1^2(q_2)\mathbf{A}_2^3(q_3)\mathbf{A}_3^4(q_4) : X_0 Y_0 Z_0 \mapsto X_4 Y_4 Z_4,$$

czyli

$$\mathbf{K}(\mathbf{q}) = \begin{bmatrix} c_{124} & -s_{124} & 0 & a_1 c_1 + a_2 c_{12} \\ s_{124} & c_{124} & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & 0 & 1 & d_1 + q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.48)$$

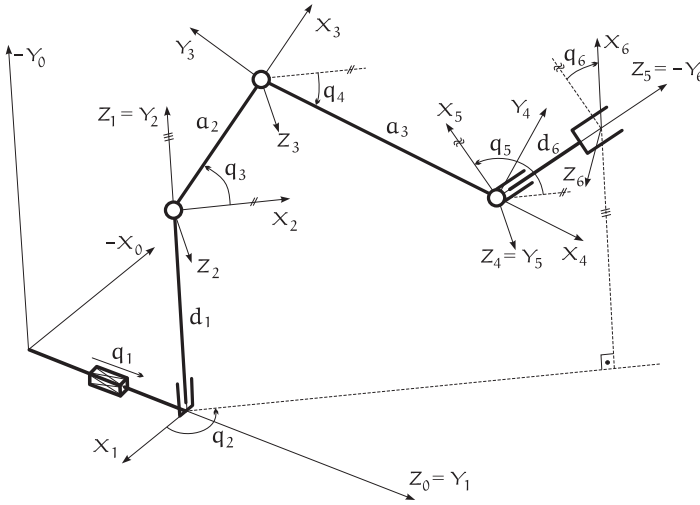
gdzie  $s_{124} = \sin(q_1 + q_2 + q_4)$  i  $c_{124} = \cos(q_1 + q_2 + q_4)$ . Wartości liczbowe parametrów manipulatora *SCARA* są równe  $d_1 = 0.8[\text{m}]$ ,  $a_1 = 0.445[\text{m}]$ ,  $a_2 = 0.355[\text{m}]$ . ■

### Przykład 2.3.4 (Manipulator IRb-6 na torze jezdnym)

Jako ostatni przykład rozpatrzmy manipulator przemysłowy IRb-6 o pięciu stopniach swobody, zamontowany na torze jezdnym<sup>¶</sup>. Taki układ tworzy manipulator klasy 1T5R<sup>||</sup> posiadający sześć stopni swobody. Schemat kinematyczny manipulatora przedstawia rysunek 2.7, a jego parametry Denavita-Hartenberga zawiera tabela 2.4. Wartości liczbowe parametrów geo-

<sup>¶</sup>Manipulator IRb-6 zamontowany na torze jezdnym znajduje się w Laboratorium Robotyki Instytutu Automatyki i Informatyki Stosowanej Politechniki Warszawskiej. Dane manipulatora udostępnił nam Prof. C. Zieliński.

<sup>||</sup>Jeden przegub przesuwny (T), po którym następuje pięć przegubów obrotowych (R).



Rysunek 2.7 Struktura kinematyczna manipulatora IRb-6 zamontowanego na torze jezdnym.

metrycznych manipulatora są następujące:  $d_1 = 0.7[\text{m}]$ ,  $a_2 = 0.45[\text{m}]$ ,  $a_3 = 0.67[\text{m}]$  i  $d_6 = 0.095[\text{m}]$ . Należy zwrócić uwagę na postać parametrów  $\theta_4$  i  $\theta_5$ . Pojawiające się w nich różnice zmiennych przegubowych wynikają ze sposobu zdefiniowania tych parametrów przez producenta manipulatora, który każdą ze zmiennych  $q_i$ ,  $i = 3, 4, 5$ , określił jako kąt między osią  $X_i$  układu współrzędnych związanego z  $i$ -tym ogniwnem manipulatora a osią  $X_{i-1}$ \*\*.

Z tabeli 2.4 wynika, że dla rozpatrywanego manipulatora przekształcenia  $A_{i-1}^i$ ,  $i = 1, \dots, 6$ , są zdefiniowane jako

$$\begin{aligned} A_0^1(q_1) &= \text{Trans}(Z, q_1) \text{Rot}(X, \frac{\pi}{2}), \\ A_1^2(q_2) &= \text{Rot}(Z, q_2) \text{Trans}(Z, d_1) \text{Rot}(X, \frac{\pi}{2}), \\ A_2^3(q_3) &= \text{Rot}(Z, q_3) \text{Trans}(X, a_2), \\ A_3^4(q_3, q_4) &= \text{Rot}(Z, q_4 - q_3) \text{Trans}(X, a_3), \\ A_4^5(q_4, q_5) &= \text{Rot}(Z, q_5 - q_4) \text{Rot}(X, \frac{\pi}{2}), \\ A_5^6(q_6) &= \text{Rot}(Z, q_6) \text{Trans}(Z, d_6) \text{Rot}(X, -\frac{\pi}{2}). \end{aligned}$$

W rezultacie, zgodnie z zależnością (2.42), kinematyka manipulatora jest

\*\*Dotychczas  $i$ -tą zmienną przegubową dla przegubów obrotowych określaliśmy jako kąt między osią  $X_i$  związaną z danym ogniwnem a osią  $X_{i-1}$ .

odwzorowaniem

$$\mathbf{K}(\mathbf{q}) = \mathbf{A}_0^1 \mathbf{A}_1^2 \mathbf{A}_2^3 \mathbf{A}_3^4 \mathbf{A}_4^5 \mathbf{A}_5^6(\mathbf{q}) : X_0 Y_0 Z_0 \mapsto X_6 Y_6 Z_6$$

postaci

$$\mathbf{K}(\mathbf{q}) = \begin{bmatrix} c_2 c_5 c_6 + s_2 s_6 & -c_2 s_5 & s_2 c_6 - c_2 c_5 s_6 \\ -s_5 c_6 & -c_5 & s_5 s_6 \\ s_2 c_5 c_6 - c_2 s_6 & -s_2 s_5 & -c_2 c_6 - s_2 c_5 s_6 \\ 0 & 0 & 0 \\ & & c_2(a_2 c_3 + a_3 c_4 + d_6 s_5) \\ & & -d_1 + d_6 c_5 - a_2 s_3 - a_3 s_4 \\ & & q_1 + s_2(a_2 c_3 + a_3 c_4 + d_6 s_5) \\ & & 1 \end{bmatrix}, \quad (2.49)$$

gdzie  $\mathbf{q} = (q_1, q_2, q_3, q_4, q_5, q_6)^T$ . ■

### 2.3.2 Reprezentacja wykładnicza

Opis kinematyki manipulatora (2.40), alternatywny w stosunku do reprezentacji Denavita-Hartenberga, można otrzymać stosując do każdej transformacji (2.41) parametryzację wykładniczą (2.28). Załóżmy najpierw, że  $i$ -ty przegub manipulatora jest obrotowy. Przy takim założeniu przekształcenie  $\mathbf{A}_{i-1}^i$  przedstawimy w następujący sposób

$$\mathbf{A}_{i-1}^i(q_i) = \begin{bmatrix} \mathbf{R}(Z, q_i) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \mathbf{P}_i, \quad (2.50)$$

gdzie

$$\mathbf{P}_i = \begin{bmatrix} \mathbb{I}_3 & d_i \mathbf{e}_3 \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}(X, \alpha_i) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbb{I}_3 & a_i \mathbf{e}_1 \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}(X, \alpha_i) & d_i \mathbf{e}_3 + a_i \mathbf{e}_1 \\ \mathbf{0} & 1 \end{bmatrix}, \quad (2.51)$$

zaś symbolem  $\mathbf{R}(\text{oś}, \text{kąt})$  oznaczamy macierz obrotu wokół ustalonej osi o zadany kąt. Nietrudno pokazać, że w parametryzacji wykładniczej

$$\begin{bmatrix} \mathbf{R}(Z, q_i) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} = \exp(\boldsymbol{\xi}_i q_i), \quad (2.52)$$

gdzie  $\boldsymbol{\xi}_i = \begin{bmatrix} \mathbf{e}_3 & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}$ . W rezultacie otrzymujemy zależność

$$\mathbf{A}_{i-1}^i(q_i) = \exp(\boldsymbol{\xi}_i q_i) \mathbf{P}_i. \quad (2.53)$$

W przypadku, gdy  $i$ -ty przegub jest przesuwny, należy wziąć

$$\xi_i = \begin{bmatrix} \mathbf{0} & \mathbf{e}_3 \\ \mathbf{0} & 0 \end{bmatrix}, \quad \mathbf{P}_i = \begin{bmatrix} \mathbf{R}(Z, \theta_i) \mathbf{R}(X, \alpha_i) & a_i \mathbf{R}(Z, \theta_i) \mathbf{1}_{kol} \\ \mathbf{0} & 1 \end{bmatrix}. \quad (2.54)$$

Podstawiając wyrażenia (2.51) lub (2.54) do wzoru (2.53) i dalej do (2.42) dochodzimy do formuły

$$\mathbf{K}(\mathbf{q}) = \exp(\xi_1 q_1) \mathbf{P}_1 \exp(\xi_2 q_2) \mathbf{P}_2 \cdots \exp(\xi_n q_n) \mathbf{P}_n. \quad (2.55)$$

Dzięki znanej własności wykładniczej funkcji macierzowej

$$\mathbf{P} \exp(\mathbf{A}) \mathbf{P}^{-1} = \exp(\mathbf{P} \mathbf{A} \mathbf{P}^{-1}),$$

równanie (2.55) możemy przepisać w postaci

$$\begin{aligned} \mathbf{K}(\mathbf{q}) = & \exp(\xi_1 q_1) \exp(\mathbf{P}_1 \xi_2 \mathbf{P}_1^{-1} q_2) \cdots \\ & \cdots \exp((\mathbf{P}_1 \cdots \mathbf{P}_{n-1}) \xi_n (\mathbf{P}_1 \cdots \mathbf{P}_{n-1})^{-1} q_n) \mathbf{P}_1 \cdots \mathbf{P}_n, \end{aligned}$$

albo jako

$$\mathbf{K}(\mathbf{q}) = \exp(\zeta_1 q_1) \exp(\zeta_2 q_2) \cdots \exp(\zeta_n q_n) \mathbf{K}(\mathbf{0}). \quad (2.56)$$

W powyższej formule

$$\zeta_1 = \xi_1 \text{ oraz } \zeta_i = \left( \prod_{l=1}^{i-1} \mathbf{P}_l \right) \xi_i \left( \prod_{l=1}^{i-1} \mathbf{P}_l \right)^{-1} \text{ dla } i = 2, \dots, n, \quad (2.57)$$

natomiast  $\mathbf{K}(\mathbf{0})$  oznacza kinematykę manipulatora w konfiguracji odniesienia, w której wszystkie zmienne przegubowe przyjmują wartość zero. Formułę (2.56) nazywamy *reprezentacją wykładniczą* kinematyki manipulatora. Macierze  $\zeta_i$  występujące w reprezentacji wykładniczej należą do algebry Liego  $\mathfrak{se}(3)$  grupy  $\mathbb{SE}(3)$  i mają postać

$$\zeta_i = \begin{bmatrix} [\boldsymbol{\omega}_i] & \mathbf{v}_i \\ \mathbf{0} & 0 \end{bmatrix}, \quad (2.58)$$

$\boldsymbol{\omega}_i, \mathbf{v}_i \in \mathbb{R}^3$ , przy czym  $\mathbf{v}_i = -\boldsymbol{\omega}_i \times \mathbf{p}_i$ , gdzie  $\mathbf{p}_i$  jest pewnym punktem położonym na osi obrotu  $i$ -tego przegubu. Wektor  $(\mathbf{v}_i^T, \boldsymbol{\omega}_i^T)^T$  reprezentujący macierz (2.58) nosi nazwę *skretnika*.



**Przykład 2.3.5 (Manipulator typu podwójne wahadło)**

Dla zilustrowania sposobu tworzenia reprezentacji wykładniczej kinematyki manipulatora posłużymy się przykładem manipulatora typu podwójne wahadło. Kinematykę tego manipulatora wyliczyliśmy w przykładzie 2.3.1 otrzymując wyrażenie

$$\mathbf{K}(\mathbf{q}) = \mathbf{A}_0^1(q_1)\mathbf{A}_1^2(q_2), \quad (2.59)$$

z macierzami  $\mathbf{A}_0^1$ ,  $\mathbf{A}_1^2$  określonymi przy pomocy formuł (2.44), (2.45). Biorąc pod uwagę strukturę tych macierzy, kinematykę (2.59) zapisujemy w następującej postaci

$$\mathbf{K}(\mathbf{q}) = \begin{bmatrix} \mathbf{R}(Z, q_1) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \mathbf{P}_1 \begin{bmatrix} \mathbf{R}(Z, q_2) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \mathbf{P}_2,$$

gdzie

$$\mathbf{P}_1 = \begin{bmatrix} \mathbb{I}_3 & l_1 \mathbf{e}_1 \\ \mathbf{0} & 1 \end{bmatrix}, \quad \mathbf{P}_2 = \begin{bmatrix} \mathbb{I}_3 & l_2 \mathbf{e}_1 \\ \mathbf{0} & 1 \end{bmatrix}.$$

Następnie, zgodnie z wyrażeniem (2.52), znajdujemy parametryzację wykładniczą macierzy obrotu

$$\mathbf{Rot}(Z, q_i) = \begin{bmatrix} \mathbf{R}(Z, q_i) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} = \exp(\boldsymbol{\xi}_i q_i), \quad i = 1, 2,$$

przy oznaczeniach  $\boldsymbol{\xi}_1 = \boldsymbol{\xi}_2 = \begin{bmatrix} \mathbf{e}_3 & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}$ . Postępując stosownie do reguł wynikających ze wzorów (2.55), (2.56), uzyskujemy reprezentację wykładniczą kinematyki (2.59)

$$\mathbf{K}(\mathbf{q}) = \exp(\boldsymbol{\zeta}_1 q_1) \exp(\boldsymbol{\zeta}_2 q_2) \mathbf{K}(\mathbf{0}), \quad (2.60)$$

scharakteryzowaną przez macierze

$$\begin{aligned} \boldsymbol{\zeta}_1 &= \boldsymbol{\xi}_1, \\ \boldsymbol{\zeta}_2 &= \mathbf{P}_1 \boldsymbol{\xi}_2 \mathbf{P}_1^{-1} = \begin{bmatrix} \mathbf{e}_3 & -l_1 \mathbf{e}_2 \\ \mathbf{0} & 0 \end{bmatrix}, \\ \mathbf{K}(\mathbf{0}) &= \begin{bmatrix} \mathbb{I}_3 & (l_1 + l_2) \mathbf{e}_1 \\ \mathbf{0} & 1 \end{bmatrix}. \end{aligned} \quad (2.61)$$

Nietrudno sprawdzić przez bezpośrednie obliczenie, że prawa strona (2.60) istotnie jest równa (2.46). ■

**Przykład 2.3.6 (Manipulator typu SCARA)**

W sposób podobny jak w poprzednim przykładzie skonstruujemy obecnie reprezentację wykładniczą kinematyki manipulatora typu SCARA, którego kinematyka

$$\mathbf{K}(\mathbf{q}) = \mathbf{A}_0^1(q_1)\mathbf{A}_1^2(q_2)\mathbf{A}_2^3(q_3)\mathbf{A}_3^4(q_4)$$

została wyznaczona w przykładzie 2.3.3. Korzystając z definicji przekształceń składowych  $\mathbf{A}_{i-1}^i$  oraz z wyrażeń (2.52) i (2.54), wyliczymy najpierw reprezentacje wykładnicze

$$\begin{aligned}\mathbf{A}_0^1(q_1) &= \begin{bmatrix} \mathbf{R}(Z, q_1) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbb{I}_3 & d_1\mathbf{e}_3 \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbb{I}_3 & a_1\mathbf{e}_1 \\ \mathbf{0} & 1 \end{bmatrix} = \exp(\boldsymbol{\xi}_1 q_1)\mathbf{P}_1, \\ \mathbf{A}_1^2(q_2) &= \begin{bmatrix} \mathbf{R}(Z, q_2) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbb{I}_3 & a_2\mathbf{e}_1 \\ \mathbf{0} & 1 \end{bmatrix} = \exp(\boldsymbol{\xi}_2 q_2)\mathbf{P}_2, \\ \mathbf{A}_2^3(q_3) &= \begin{bmatrix} \mathbb{I}_3 & q_3\mathbf{e}_3 \\ \mathbf{0} & 1 \end{bmatrix} = \exp(\boldsymbol{\xi}_3 q_3)\mathbf{P}_3, \\ \mathbf{A}_3^4(q_4) &= \begin{bmatrix} \mathbf{R}(Z, q_4) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} = \exp(\boldsymbol{\xi}_4 q_4)\mathbf{P}_4,\end{aligned}\tag{2.62}$$

gdzie

$$\begin{aligned}\boldsymbol{\xi}_1 = \boldsymbol{\xi}_2 = \boldsymbol{\xi}_4 &= \begin{bmatrix} [\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}, \quad \boldsymbol{\xi}_3 = \begin{bmatrix} \mathbf{0} & \mathbf{e}_3 \\ \mathbf{0} & 0 \end{bmatrix}, \\ \mathbf{P}_1 &= \begin{bmatrix} \mathbb{I}_3 & a_1\mathbf{e}_1 + d_1\mathbf{e}_3 \\ \mathbf{0} & 1 \end{bmatrix}, \quad \mathbf{P}_2 = \begin{bmatrix} \mathbb{I}_3 & a_2\mathbf{e}_1 \\ \mathbf{0} & 1 \end{bmatrix}, \quad \mathbf{P}_3 = \mathbf{P}_4 = \mathbb{I}_4.\end{aligned}$$

Na podstawie formuł (2.56), (2.57) i po uwzględnieniu postaci reprezentacji (2.62), otrzymujemy ostatecznie

$$\mathbf{K}(\mathbf{q}) = \exp(\boldsymbol{\zeta}_1 q_1) \exp(\boldsymbol{\zeta}_2 q_2) \exp(\boldsymbol{\zeta}_3 q_3) \exp(\boldsymbol{\zeta}_4 q_4)\mathbf{K}(\mathbf{0}),\tag{2.63}$$

przy oznaczeniach

$$\begin{aligned}\boldsymbol{\zeta}_1 = \boldsymbol{\xi}_1, \quad \boldsymbol{\zeta}_2 &= \mathbf{P}_1\boldsymbol{\xi}_2\mathbf{P}_1^{-1} = \begin{bmatrix} [\mathbf{e}_3] & -a_1\mathbf{e}_2 \\ \mathbf{0} & 0 \end{bmatrix}, \quad \boldsymbol{\zeta}_3 = (\mathbf{P}_1\mathbf{P}_2)\boldsymbol{\xi}_3(\mathbf{P}_1\mathbf{P}_2)^{-1} = \boldsymbol{\xi}_3, \\ \boldsymbol{\zeta}_4 &= (\mathbf{P}_1\mathbf{P}_2\mathbf{P}_3)\boldsymbol{\xi}_4(\mathbf{P}_1\mathbf{P}_2\mathbf{P}_3)^{-1} = \begin{bmatrix} [\mathbf{e}_3] & -(a_1 + a_2)\mathbf{e}_2 \\ \mathbf{0} & 0 \end{bmatrix}, \\ \mathbf{K}(\mathbf{0}) &= \mathbf{P}_1\mathbf{P}_2\mathbf{P}_3\mathbf{P}_4 = \begin{bmatrix} \mathbb{I}_3 & (a_1 + a_2)\mathbf{e}_1 + d_1\mathbf{e}_3 \\ \mathbf{0} & 1 \end{bmatrix}.\end{aligned}\tag{2.64}$$

■

### 2.3.3 Kinematyka we współrzędnych

Odwzorowanie (2.42) zostało określone między rozmaitościami: przegubową a zadaniową. Używając naturalnych współrzędnych  $\varphi_{\mathbf{U}} : \mathbf{U} \subset \mathcal{Q} \rightarrow \mathbb{R}^n$  na rozmaitości przegubowej oraz wybierając pewną parametryzację  $\psi_{\mathbf{V}} : \mathbf{V} \subset \mathbb{R}^m \rightarrow \mathcal{Z}$  rozmaitości zadaniowej otrzymujemy *reprezentację kinematyki we współrzędnych*

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

zapewniającą przemienność diagramu

$$\begin{array}{ccc} \mathcal{Q} & \xrightarrow{\mathbf{K}} & \mathcal{Z} \\ \cup & & \parallel \\ \mathbf{U} & \xrightarrow{\mathbf{K}|_{\mathbf{U}}} & \mathcal{Z} \\ \varphi_{\mathbf{U}} \downarrow & & \uparrow \psi_{\mathbf{V}} \\ \mathbb{R}^n & \xrightarrow{\mathbf{k}} & \mathbf{V} \subset \mathbb{R}^m, \end{array}$$

a zatem spełniającą warunek

$$\mathbf{K}|_{\mathbf{U}} = \psi_{\mathbf{V}} \circ \mathbf{k} \circ \varphi_{\mathbf{U}}. \quad (2.65)$$

Ze struktury diagramu wynika, że wyznaczenie reprezentacji kinematyki we współrzędnych sprowadza się do przeprowadzenia faktoryzacji kinematyki  $\mathbf{K}|_{\mathbf{U}}$  i nadaniu jej formy złożenia trzech odwzorowań  $\varphi_{\mathbf{U}}$ ,  $\mathbf{k}$  i  $\psi_{\mathbf{V}}$ . Lokalnie, w obszarze odwracalności odwzorowań  $\psi_{\mathbf{V}}$ ,  $\varphi_{\mathbf{U}}$ , reprezentację  $\mathbf{k}$  można wyliczyć jako  $\mathbf{k} = \psi_{\mathbf{V}}^{-1} \circ \mathbf{K}|_{\mathbf{U}} \circ \varphi_{\mathbf{U}}^{-1}$ . Współrzędne  $\mathbf{x}$  nazywamy *konfiguracyjnymi* lub *przegubowymi*, współrzędne  $\mathbf{y}$  — *zadaniowymi*. Przestrzeń  $\mathbb{R}^n$  nazywa się *przestrzenią konfiguracyjną* lub *przestrzenią przegubową*, a przestrzeń  $\mathbb{R}^m$  — *przestrzenią zadaniową*. Jeżeli liczba stopni swobody manipulatora jest większa od wymiaru jego przestrzeni zadaniowej, to manipulator nazywamy *redundantnym*, w przeciwnym przypadku manipulator nazywamy *nieredundantnym*. Nadmiar możliwości ruchowych, jakim dysponuje manipulator redundantny, pozwala sterować jakością wykonania zadań przez manipulator.

Przedstawimy teraz kilka przykładów wyliczania kinematyki manipulatora we współrzędnych.

**Przykład 2.3.7 (Manipulator typu podwójne wahadło)**

Zacznijmy od wyrażenia opisującego kinematykę manipulatora typu podwójne wahadło we współrzędnych. Jak wiemy, kąty obrotu w przegubach manipulatora  $x_1$  i  $x_2$  opisują jednoznacznie jego pozycję i tworzą dwuelementowy wektor współrzędnych przegubowych

$$\mathbf{x} = (x_1, x_2)^T \in \mathbb{R}^2.$$

Ponieważ pozycje efektora manipulatora odpowiadają punktom płaszczyzny  $X_0Y_0$ , do opisu położenia i orientacji efektora użyjemy grupy euklidesowej  $SE(2) \cong \mathbb{R}^2 \times S^1$ , która stanowi rozmaitość zadaniową  $\mathcal{Z}$ . Jako współrzędne zadaniowe przyjmijmy, odpowiednio, położenie efektora wzdłuż osi  $X_0, Y_0$  efektora oraz jego orientację  $\varphi$  mierzona kątem obrotu wokół osi  $Z_0$  (zobacz rysunek 2.4)

$$\begin{cases} y_1 = X_0 \\ y_2 = Y_0 \\ y_3 = \varphi. \end{cases}$$

Dla tak wybranych współrzędnych zdefiniujemy przekształcenie

$$\mathbf{Trans}(X, y_1)\mathbf{Trans}(Y, y_2)\mathbf{Rot}(Z, y_3) = \begin{bmatrix} \cos y_3 & -\sin y_3 & 0 & y_1 \\ \sin y_3 & \cos y_3 & 0 & y_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.66)$$

Porównując wyrażenie (2.66) z kinematyką  $\mathbf{K}$  daną równaniem (2.46), otrzymujemy reprezentację kinematyki we współrzędnych

$$\mathbf{k} : \mathbb{R}^2 \longrightarrow \mathbb{R}^3,$$

w postaci układu równań

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} l_1 c_1 + l_2 c_{12} \\ l_1 s_1 + l_2 s_{12} \\ x_1 + x_2 \end{pmatrix}, \quad (2.67)$$

gdzie  $s_i, c_i, s_{ij}$  i  $c_{ij}$  oznaczają, odpowiednio,  $\sin x_i, \cos x_i, \sin(x_i + x_j)$  oraz  $\cos(x_i + x_j)$ .

Ponieważ rozpatrywany manipulator jest wyposażony w dwa przeguby, jedynie dwie spośród wyliczonych trzech współrzędnych zadaniowych można

zmieniać niezależnie poprzez zmianę współrzędnych przegubowych\*. Z tego powodu częstokroć definiuje się kinematykę manipulatora ograniczoną jedynie do dwóch współrzędnych — mówimy wtedy o kinematyce w *aspekcie* tych współrzędnych. I tak, wybierając z wyrażenia (2.67)<sup>†</sup> dwie pierwsze składowe otrzymujemy kinematykę manipulatora w aspekcie współrzędnych  $x_0 y_0$

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} l_1 c_1 + l_2 c_{12} \\ l_1 s_1 + l_2 s_{12} \end{pmatrix}. \quad (2.68)$$

Podobnie jak poprzednio mamy  $y_1 = x_0$  a  $y_2 = y_0$ . Oczywiście, taki wybór współrzędnych zadaniowych nie jest jedyny. Alternatywnie, można spośród współrzędnych zadaniowych wybrać np. ponownie współrzędną  $x_0$  położenia efektora w podstawowym układzie współrzędnych  $x_0 y_0 z_0$  oraz kąt  $\varphi$  określający orientację efektora względem osi  $Z_0$  tego układu. W takim przypadku dostajemy kinematykę w aspekcie współrzędnych  $x_0 \varphi$

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} l_1 c_1 + l_2 c_{12} \\ x_1 + x_2 \end{pmatrix}, \quad (2.69)$$

gdzie  $y_1 = x_0$  oraz  $y_2 = \varphi$ . ■

### Przykład 2.3.8 (Manipulator *EDDA*)

Jak zauważyliśmy wcześniej, kinematykę podwójnego wahadła posiada manipulator *EDDA* (Experimental Direct Drive Arm)<sup>‡</sup>. *EDDA* jest manipulatorem o bezpośrednim napędzie i nieograniczonych przegubach. Jej konstrukcję przedstawia rysunek 2.8. W zależności od wybranego aspektu kinematyka manipulatora *EDDA* jest opisana jednym z równań (2.67)–(2.69) z parametrami  $l_1 = 0.3[\text{m}]$ ,  $l_2 = 0.58[\text{m}]$ . ■

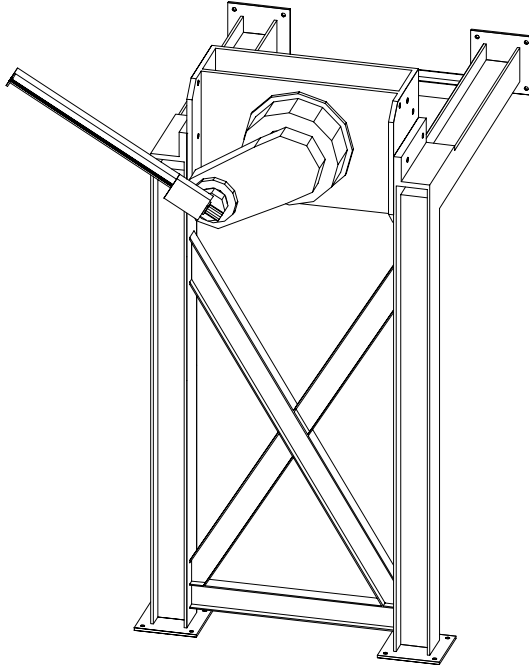
### Przykład 2.3.9 (Manipulator typu potrójne wahadło)

Przyjmując  $\mathbf{x} = (x_1, x_2, x_3)^T$  jako wektor współrzędnych na rozmaitości konfiguracyjnej  $\mathcal{Q}$ , gdzie  $x_i$  ( $i = 1, 2, 3$ ) oznaczają kąty obrotu w kolejnych przegubach manipulatora, oraz wybierając położenie efektora wzdłuż osi  $x_0$

\* Jak się później okaże (zobacz podrozdział 2.3.5), manipulator z tak zdefiniowaną kinematyką pozostaje zawsze osobliwy.

<sup>†</sup> Które może być uważane za kinematykę w aspekcie współrzędnych  $x_0 y_0 \varphi$ .

<sup>‡</sup> Manipulator *EDDA* znajduje się w Instytucie Robotyki i Informatyki Uniwersytetu w Brunzshwiku. Za jego udostępnienie jesteśmy wdzięczni Prof. F. Wahlowi.

Rysunek 2.8 Manipulator *EDDA*.

i  $\gamma_0$ , i jego orientację  $\varphi$  mierzoną kątem obrotu wokół osi  $Z_0$  jako parametryzację  $y_1$ ,  $y_2$  i  $y_3$  rozmaitości zadaniowej  $\text{SE}(2)$ <sup>§</sup>, otrzymujemy kinematykę manipulatora typu potrójne wahadło w aspekcie wybranych współrzędnych

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} l_1 c_1 + l_2 c_{12} + l_3 c_{123} \\ l_1 s_1 + l_2 s_{12} + l_3 s_{123} \\ x_1 + x_2 + x_3 \end{pmatrix}. \quad (2.70)$$

Często manipulator typu potrójne wahadło jest traktowany jako manipulator redundantny z kinematyką wyrażoną w aspekcie współrzędnych  $X_0 Y_0$  postaci

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} l_1 c_1 + l_2 c_{12} + l_3 c_{123} \\ l_1 s_1 + l_2 s_{12} + l_3 s_{123} \end{pmatrix}. \quad (2.71)$$

■

<sup>§</sup> Jak w przypadku manipulatora typu podwójne wahadło, nie ma potrzeby reprezentowania całej grupy  $\text{SE}(3)$ .

**Przykład 2.3.10 (Manipulator typu SCARA)**

W celu wyrażania we współrzędnych kinematyki manipulatora typu *SCARA* wykorzystamy naturalne współrzędne przegubowe manipulatora. Ponieważ położenia przegubów manipulatora wpływają jedynie na położenie efektora w przestrzeni  $\mathbb{R}^3$  oraz na jedną składową jego orientacji wyrażoną kątem obrotu wokół osi  $Z_0$ , wybierzemy cztery współrzędne zadaniowe manipulatora: współrzędne  $(y_1, y_2$  i  $y_3)$  określające położenie efektora manipulatora względem podstawowego układu współrzędnych  $X_0Y_0Z_0$  oraz współrzędną  $(y_4)$  w postaci kąta obrotu efektora manipulatora wokół osi  $Z_0$  układu podstawowego. Przy tak wybranych współrzędnych otrzymujemy przekształcenie

$$\begin{aligned} \mathbf{Trans}(X, y_1)\mathbf{Trans}(Y, y_2)\mathbf{Trans}(Z, y_3)\mathbf{Rot}(Z, y_4) = \\ = \begin{bmatrix} \cos y_4 & -\sin y_4 & 0 & y_1 \\ \sin y_4 & \cos y_4 & 0 & y_2 \\ 0 & 0 & 1 & y_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \end{aligned}$$

którego porównanie z kinematyką zdefiniowaną równaniem (2.48) pozwala uzyskać następującą reprezentację

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} a_1c_1 + a_2c_{12} \\ a_1s_1 + a_2s_{12} \\ d_1 + x_3 \\ x_1 + x_2 + x_4 \end{pmatrix}. \quad (2.72)$$

W sytuacji, gdy do wykonania zadania potrzebna jest wyłącznie znajomość położenia efektora manipulatora *SCARA*, kinematykę

$$\mathbf{k} : \mathbb{R}^4 \longrightarrow \mathbb{R}^3$$

manipulatora w aspekcie współrzędnych  $X_0Y_0Z_0$  definiujemy jako

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} a_1c_1 + a_2c_{12} \\ a_1s_1 + a_2s_{12} \\ d_1 + x_3 \end{pmatrix}. \quad (2.73)$$

Ponieważ pełna specyfikacja kinematyki manipulatora przemysłowego wymaga określenia zakresu dopuszczalnych ruchów w przegubach, na zakończenie dodajmy, że przestrzeń przegubowa manipulatora typu *SCARA*

jest ograniczona w następujący sposób:

$$\mathcal{X} = \{ \mathbf{x} = (x_1, \dots, x_4)^T \in \mathbb{R}^4 \mid -\frac{1}{2}\pi \leq x_1 \leq \frac{1}{2}\pi, -\frac{3}{4}\pi \leq x_2 \leq \frac{3}{4}\pi, \\ 0.15[\text{m}] \leq x_3 \leq 0.39[\text{m}], -\pi \leq x_4 \leq \pi \}. \quad (2.74)$$

■

### Przykład 2.3.11 (Manipulator IRb-6 na torze jezdny)

Aby wyrazić we współrzędnych kinematykę manipulatora IRb-6 zamontowanego na torze jezdny, rozmaitość zadaniową sparametryzujemy przy pomocy trzech współrzędnych położenia efektora manipulatora ( $y_1, y_2, y_3$ ) oraz trzech kątów Eulera Y-Z-Y ( $y_4, y_5, y_6$ ). Na rozmaitości konfiguracyjnej użyjemy naturalnego układu współrzędnych zadanego przez zmienne przegubowe manipulatora. Dla tak zdefiniowanych współrzędnych otrzymujemy przekształcenie

$$\text{Trans}(X, y_1) \text{Trans}(Y, y_2) \text{Trans}(Z, y_3) \text{Rot}(Y, y_4) \text{Rot}(Z, y_5) \text{Rot}(Y, y_6) = \\ = \begin{bmatrix} \check{c}_4 \check{c}_5 \check{c}_6 - \check{s}_4 \check{s}_6 & -\check{c}_4 \check{s}_5 & \check{s}_4 \check{c}_6 + \check{c}_4 \check{c}_5 \check{s}_6 & y_1 \\ \check{s}_5 \check{c}_6 & \check{c}_5 & \check{s}_5 \check{s}_6 & y_2 \\ -\check{s}_4 \check{c}_5 \check{c}_6 - \check{c}_4 \check{s}_6 & \check{s}_4 \check{s}_5 & \check{c}_4 \check{c}_6 - \check{s}_4 \check{c}_5 \check{s}_6 & y_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.75)$$

gdzie przez  $\check{s}_i$  i  $\check{c}_i$  oznaczono  $\sin y_i$  i  $\cos y_i$ . Porównując ze sobą macierze (2.49) i (2.75) otrzymujemy reprezentację kinematyki manipulatora we współrzędnych

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{pmatrix} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} c_2(a_2c_3 + a_3c_4 + d_6s_5) \\ -d_1 + d_6c_5 - a_2s_3 - a_3s_4 \\ x_1 + s_2(a_2c_3 + a_3c_4 + d_6s_5) \\ -x_2 \\ \pi - x_5 \\ \pi - x_6 \end{pmatrix}. \quad (2.76)$$

Dodajmy, że wybór współrzędnych do reprezentacji kinematyki ma istotny wpływ na złożoność równań kinematyki.

W dalszych przykładach będziemy niekiedy traktować manipulator IRb-6 na torze jezdny jako manipulator redundantny, którego kinematyka

$$\mathbf{k} : \mathbb{R}^6 \longrightarrow \mathbb{R}^3$$



opisuje położenie efektora manipulatora w aspekcie współrzędnych  $x_0 y_0 z_0$  i jest postaci

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} c_2(a_2c_3 + a_3c_4 + d_6s_5) \\ -d_1 + d_6c_5 - a_2s_3 - a_3s_4 \\ x_1 + s_2(a_2c_3 + a_3c_4 + d_6s_5) \end{pmatrix}. \quad (2.77)$$

W uzupełnieniu dodajmy, że przestrzeń przegubowa manipulatora IRb-6 zamontowanego na torze jezdny jest ograniczona przez jego konstrukcję mechaniczną w następujący sposób:

$$\begin{aligned} \mathcal{X} = \{ \mathbf{x} = (x_1, \dots, x_6)^T \in \mathbb{R}^6 \mid & 0[\text{m}] \leq x_1 \leq 1.5[\text{m}], \\ & -\frac{17}{18}\pi \leq x_2 \leq \frac{17}{18}\pi, \frac{5}{18}\pi \leq x_3 \leq \frac{13}{18}\pi, -\frac{2}{9}\pi \leq x_4 \leq \frac{5}{36}\pi, \\ & 0 \leq x_5 \leq \pi, -\pi \leq x_6 \leq \pi, \frac{5}{18}\pi \leq x_3 - x_4 \leq \frac{13}{18}\pi \}. \end{aligned} \quad (2.78)$$

■

### 2.3.4 Jakobiany

Macierz Jacobiego

$$\mathbf{J}^a(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}), \quad (2.79)$$

reprezentacji kinematyki manipulatora we współrzędnych

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

nazywamy *jakobianem analitycznym* manipulatora. Jakobian analityczny można traktować jako przekształcenie prędkości zmian współrzędnych przegubowych w prędkość zmian współrzędnych zadaniowych, przy zadanej konfiguracji manipulatora,

$$\dot{\mathbf{y}} = \mathbf{J}^a(\mathbf{x})\dot{\mathbf{x}}. \quad (2.80)$$

Jest oczywiste, że postać jakobianu analitycznego zależy od wyboru układu współrzędnych (parametryzacji) dokonanego w celu zdefiniowania odwzorowania  $\mathbf{k}$ .

**Przykład 2.3.12 (Manipulator typu podwójne wahadło)**

Różniczkując równania kinematyki (2.67) otrzymujemy jacobian analityczny

$$\mathbf{J}^a(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}) = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} & -l_2 s_{12} \\ l_1 c_1 + l_2 c_{12} & l_2 c_{12} \\ 1 & 1 \end{bmatrix}, \quad (2.81)$$

gdzie  $s_i$ ,  $c_i$ ,  $s_{ij}$  i  $c_{ij}$  oznaczają, odpowiednio,  $\sin x_i$ ,  $\cos x_i$ ,  $\sin(x_i + x_j)$  oraz  $\cos(x_i + x_j)$ \*. Podobnie jak w przypadku kinematyki wyrażonej we współrzędnych, możemy mówić o jacobianach analitycznych w aspekcie różnych współrzędnych. Wyliczony wyżej jacobian jest jacobianem analitycznym w aspekcie współrzędnych  $x_0 y_0 \varphi$ . Przez analogię, dla kinematyki zdefiniowanej wzorem (2.69) otrzymujemy jacobian analityczny w aspekcie współrzędnych  $x_0 y_0$

$$\mathbf{J}^a(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}) = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} & -l_2 s_{12} \\ l_1 c_1 + l_2 c_{12} & l_2 c_{12} \end{bmatrix}. \quad (2.82)$$

Jacobian analityczny w aspekcie współrzędnych  $x_0 \varphi$  odpowiadający kinematyce (2.69) ma postać

$$\mathbf{J}^a(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}) = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} & -l_2 s_{12} \\ 1 & 1 \end{bmatrix}. \quad (2.83)$$

■

**Przykład 2.3.13 (Manipulator typu potrójne wahadło)**

Bezpośrednie zróżniczkowanie kinematyki (2.70) potrójnego wahadła pozwala wyliczyć jacobian analityczny jako macierz

$$\mathbf{J}^a(\mathbf{x}) = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} - l_3 s_{123} & -l_2 s_{12} - l_3 s_{123} & -l_3 s_{123} \\ l_1 c_1 + l_2 c_{12} + l_3 c_{123} & l_2 c_{12} + l_3 c_{123} & l_3 c_{123} \\ 1 & 1 & 1 \end{bmatrix}. \quad (2.84)$$

W podobny sposób, dla kinematyki (2.71) otrzymujemy

$$\mathbf{J}^a(\mathbf{x}) = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} - l_3 s_{123} & -l_2 s_{12} - l_3 s_{123} & -l_3 s_{123} \\ l_1 c_1 + l_2 c_{12} + l_3 c_{123} & l_2 c_{12} + l_3 c_{123} & l_3 c_{123} \end{bmatrix}. \quad (2.85)$$

■

---

\*Przyjmujemy oznaczenia funkcji trygonometrycznych zmiennych  $q_i$  i  $x_i$  takie same jak w formule kinematyki. Z kontekstu zawsze będzie jasno wynikało, o funkcjach których zmiennych mówimy — przez  $s_i$ ,  $c_i$  rozumiemy odpowiednio  $\sin x_i$  i  $\cos x_i$  w przypadku jacobianów analitycznych, oraz  $\sin q_i$  i  $\cos q_i$  dla pozostałych typów jacobianów.

**Przykład 2.3.14 (Manipulator typu SCARA)**

Kinematyce (2.72) manipulatora typu SCARA odpowiada jacobian analityczny

$$\mathbf{J}^a(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}) = \begin{bmatrix} -a_1 s_1 - a_2 s_{12} & -a_2 s_{12} & 0 & 0 \\ a_1 c_1 + a_2 c_{12} & a_2 c_{12} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}. \quad (2.86)$$

Jest rzeczą jasną, że kinematyce (2.73) rozpatrywanej w aspekcie współrzędnych  $X_0 Y_0 Z_0$  odpowiada jacobian analityczny uzyskany przez wykreślenie ostatniego wiersza macierzy (2.86). ■

**Przykład 2.3.15 (Manipulator IRb-6 na torze jezdny)**

Na podstawie równań we współrzędnych (2.76) kinematyki manipulatora IRb-6 zamontowanego na torze jezdny obliczamy jacobian analityczny

$$\mathbf{J}^a(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}) = \begin{bmatrix} 0 & -s_2(a_2 c_3 + a_3 c_4 + d_6 s_5) & -a_2 c_2 s_3 & -a_3 c_2 s_4 & d_6 c_2 c_5 & 0 \\ 0 & 0 & -a_2 c_3 & -a_3 c_4 & -d_6 s_5 & 0 \\ 1 & c_2(a_2 c_3 + a_3 c_4 + d_6 s_5) & -a_2 s_2 s_3 & -a_3 s_2 s_4 & d_6 s_2 c_5 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}. \quad (2.87)$$

Łatwo zauważyć, że kinematyce (2.77) odpowiada jacobian analityczny złożony z trzech pierwszych wierszy macierzy (2.87). ■

Jacobian manipulatora może być także zdefiniowany bez odwoływania się do układów współrzędnych. Wykorzystując fakt, że kinematyka manipulatora przyjmuje wartości w specjalnej grupie euklidesowej<sup>†</sup>  $\mathbb{SE}(3)$ , wprowadza się niezależne od współrzędnych pojęcie *jacobianu geometrycznego*, a dokładniej: jacobianu geometrycznego w układzie przestrzeni i jacobianu geometrycznego w układzie ciała, zwanych w skrócie jacobianami geometrycznymi w przestrzeni i w ciele. Jacobian geometryczny w przestrzeni jest rozumiany jako przekształcenie prędkości ruchu w przegubach w wektor prędkości efektora w przestrzeni

$$\begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix} = \mathbf{J}^s(\mathbf{q})\dot{\mathbf{q}}. \quad (2.88)$$

<sup>†</sup>A więc w grupie Liego z dobrze określoną przestrzenią styczną w każdym punkcie.

Analogicznie, jacobian geometryczny w ciele stanowi przekształcenie prędkości ruchu przegubów w wektor prędkości efektora w ciele,

$$\begin{pmatrix} \mathbf{v}_b \\ \boldsymbol{\omega}_b \end{pmatrix} = \mathbf{J}^b(\mathbf{q})\dot{\mathbf{q}}. \quad (2.89)$$

Przedstawimy teraz algorytm wyliczania jacobianu geometrycznego w przestrzeni. Niech

$$\mathbf{J}^s(\mathbf{q}) = [\mathbf{J}_1^s(\mathbf{q}) \quad \cdots \quad \mathbf{J}_i^s(\mathbf{q}) \quad \cdots \quad \mathbf{J}_n^s(\mathbf{q})].$$

W celu wyznaczenia  $i$ -tej kolumny jacobianu  $\mathbf{J}^s(\mathbf{q})$  założmy, że wszystkie przeguby manipulatora, oprócz przegubu  $i$ -tego, zostały unieruchomione i przeanalizujemy ruch efektora będący rezultatem ruchu w przegubie o numerze  $i$ . Niech położenie efektora względem układu  $X_{i-1}Y_{i-1}Z_{i-1}$  będzie opisane wektorem  $\mathbf{T}_{i-1}^n$ , a przegub  $i$  będzie typu obrotowego. Wówczas, z zależności (2.12) wynika, że prędkości efektora w układzie przestrzeni względem układu  $(i-1)$ -szego wynoszą

$$\boldsymbol{\omega}_s^{i-1} = \mathbf{e}_3\dot{q}_i, \quad \mathbf{v}_s^{i-1} = \dot{\mathbf{T}}_{i-1}^n(q_i) - [\boldsymbol{\omega}_s^{i-1}] \mathbf{T}_{i-1}^n(q_i). \quad (2.90)$$

Zajmiemy się teraz wyliczeniem prędkości  $\dot{\mathbf{T}}_{i-1}^n(q_i)$ . Niech

$$\mathbf{A}_{i-1}^i(q_i) = \begin{bmatrix} \mathbf{R}_{i-1}^i(q_i) & \mathbf{T}_{i-1}^i(q_i) \\ \mathbf{0} & \mathbf{1} \end{bmatrix}$$

oznacza transformację układu  $(i-1)$ -szego w układ  $i$ -ty, zdefiniowaną przy pomocy algorytmu Denavita-Hartenberga. Jeżeli położenie efektora względem układu  $i$ -tego oznaczmy przez  $\mathbf{T}_i^n$ , to

$$\mathbf{T}_{i-1}^n(q_i) = \mathbf{R}_{i-1}^i(q_i)\mathbf{T}_i^n + \mathbf{T}_{i-1}^i(q_i). \quad (2.91)$$

Zróżniczkowanie (2.91) względem czasu daje

$$\dot{\mathbf{T}}_{i-1}^n(q_i) = [\mathbf{e}_3]\mathbf{R}_{i-1}^i(q_i)\mathbf{T}_i^n\dot{q}_i + \dot{\mathbf{T}}_{i-1}^i(q_i). \quad (2.92)$$

Wektor  $\mathbf{T}_{i-1}^i(q_i)$  wyliczamy korzystając z wyrażenia (2.51)

$$\mathbf{T}_{i-1}^i(q_i) = a_i\mathbf{R}(Z, q_i)\mathbf{e}_1 + d_i\mathbf{e}_3. \quad (2.93)$$

Biorąc pod uwagę (2.93) i pamiętając, że  $[\mathbf{e}_3]\mathbf{e}_3 = 0$  obliczamy

$$\dot{\mathbf{T}}_{i-1}^i(q_i) = [\mathbf{e}_3]\dot{q}_i a_i\mathbf{R}(Z, q_i)\mathbf{e}_1 = [\mathbf{e}_3]\dot{q}_i\mathbf{T}_{i-1}^i(q_i). \quad (2.94)$$

W końcu, łącząc (2.94) i (2.92) oraz wykorzystując (2.91) zauważamy, że

$$\dot{\mathbf{T}}_{i-1}^n(\mathbf{q}_i) = [\mathbf{e}_3] \dot{q}_i (\mathbf{R}_{i-1}^i \mathbf{T}_i^n + \mathbf{T}_{i-1}^i) = [\boldsymbol{\omega}_s^{i-1}] \mathbf{T}_{i-1}^n(\mathbf{q}_i),$$

a zatem

$$\mathbf{v}_s^{i-1} = \mathbf{0}. \quad (2.95)$$

Po obliczeniu prędkości w przestrzeni względem układu  $(i-1)$ -szego, wyliczamy prędkości w przestrzeni względem układu podstawowego korzystając z zależności

$$\begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix} = \begin{bmatrix} \mathbf{R}_0^{i-1} & [\mathbf{T}_0^{i-1}] \mathbf{R}_0^{i-1} \\ \mathbf{0} & \mathbf{R}_0^{i-1} \end{bmatrix} \begin{pmatrix} \mathbf{v}_s^{i-1} \\ \boldsymbol{\omega}_s^{i-1} \end{pmatrix}, \quad (2.96)$$

gdzie

$$\begin{bmatrix} \mathbf{R}_0^{i-1}(\mathbf{q}^{i-1}) & \mathbf{T}_0^{i-1}(\mathbf{q}^{i-1}) \\ \mathbf{0} & \mathbf{1} \end{bmatrix} = \prod_{k=1}^{i-1} \mathbf{A}_{k-1}^k(\mathbf{q}_k), \quad (2.97)$$

przy oznaczeniu  $\mathbf{q}^{i-1} = (q_1, \dots, q_{i-1})^T$ , przedstawia transformację układu podstawowego w układ  $(i-1)$ -szy. Dla  $i=1$ , w powyższym wzorze należy przyjąć  $\mathbf{R}_0^0 = \mathbb{I}_3$ ,  $\mathbf{T}_0^0 = \mathbf{0}$ . Z zależności (2.90) i (2.95) otrzymujemy następującą postać  $i$ -tej kolumny jacobianu geometrycznego w przestrzeni odpowiadającej przegubowi obrotowemu manipulatora

$$\mathbf{J}_i^s(\mathbf{q}) = \begin{pmatrix} \mathbf{T}_0^{i-1}(\mathbf{q}^{i-1}) \times \mathbf{R}_{03\text{kol}}^{i-1}(\mathbf{q}^{i-1}) \\ \mathbf{R}_{03\text{kol}}^{i-1}(\mathbf{q}^{i-1}) \end{pmatrix}. \quad (2.98)$$

W przypadku gdy  $i$ -ty przegub manipulatora jest przesuwany, podobne rozumowanie prowadzi do wniosku, że  $\boldsymbol{\omega}_s^{i-1} = \mathbf{0}$ ,  $\mathbf{v}_s^{i-1} = \mathbf{e}_3 \dot{q}_i$ , a zatem

$$\mathbf{J}_i^s(\mathbf{q}) = \begin{pmatrix} \mathbf{R}_{03\text{kol}}^{i-1}(\mathbf{q}^{i-1}) \\ \mathbf{0} \end{pmatrix}. \quad (2.99)$$

We wzorach (2.98), (2.99) symbol  $\mathbf{R}_{03\text{kol}}^{i-1}(\mathbf{q}^{i-1})$  oznacza trzecią kolumnę macierzy  $\mathbf{R}_0^{i-1}(\mathbf{q}^{i-1})$  występującej w transformacji (2.97).

### Przykład 2.3.16 (Manipulator typu podwójne wahadło)

Obecnie wyznaczmy jacobian geometryczny w przestrzeni dla manipulatora typu podwójne wahadło. Ponieważ ten manipulator jest wyposażony wyłącznie w przeguby obrotowe, do wyznaczenia jacobianu posłużymy się

zależnością (2.98). Dla pierwszego przegubu manipulatora dostajemy od razu

$$\mathbf{J}_1^s = (0, 0, 0, 0, 0, 1)^T. \quad (2.100)$$

Do wyliczenia kolumny jacobianu geometrycznego w przestrzeni wykorzystamy macierz  $\mathbf{A}_0^1$  daną zależnością (2.44). W rezultacie otrzymujemy

$$\mathbf{J}_2^s(q_1) = (l_1 \sin q_1, -l_1 \cos q_1, 0, 0, 0, 1)^T. \quad (2.101)$$

Jakobian geometryczny w przestrzeni manipulatora typu podwójne wahadło składa się z kolumn (2.100), (2.101)

$$\mathbf{J}^s(\mathbf{q}) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ l_1 s_1 & -l_1 c_1 & 0 & 0 & 0 & 1 \end{bmatrix}^T. \quad (2.102)$$

■

### Przykład 2.3.17 (Manipulator typu potrójne wahadło)

Dla manipulatora typu potrójne wahadło, na podstawie zależności (2.98), wyznaczamy jacobian geometryczny w przestrzeni w postaci

$$\mathbf{J}^s(\mathbf{q}) = \begin{bmatrix} 0 & l_1 s_1 & l_1 s_1 + l_2 s_{12} \\ 0 & -l_1 c_1 & -l_1 c_1 - l_2 c_{12} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}. \quad (2.103)$$

■

### Przykład 2.3.18 (Manipulator typu SCARA)

Jacobian geometryczny w przestrzeni manipulatora typu SCARA znajdujemy przy wykorzystaniu zależności (2.98) i (2.99)<sup>‡</sup>. Ma on następującą postać

$$\mathbf{J}^s(\mathbf{q}) = \begin{bmatrix} 0 & a_1 s_1 & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & -a_1 c_1 & 0 & -a_1 c_1 - a_2 c_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}. \quad (2.104)$$

■

<sup>‡</sup>Poszczególne kolumny tego jacobianu wylicza się, odpowiednio, na podstawie macierzy  $\mathbf{A}_0^0 = \mathbb{I}_4$ ,  $\mathbf{A}_0^1$ ,  $\mathbf{A}_0^2$ ,  $\mathbf{A}_0^3$ .

**Przykład 2.3.19 (Manipulator IRb-6 na torze jezdny)**

Wyliczmy teraz jacobian geometryczny w przestrzeni  $\mathbf{J}^s$  manipulatora IRb-6 zamontowanego na torze jezdny. Dla przegubu przesuwneego pierwsza kolumna jacobianu jest zawsze równa  $\mathbf{J}_1^s(\mathbf{q}) = (0, 0, 1, 0, 0, 0)^\top$ . Zauważmy, że zależności (2.98) i (2.99) zostały wyprowadzone przy założeniu, że zmianie pojedynczej zmiennej przegubowej towarzyszy ruch w jednym przegubie manipulatora. Niestety, w naszym przykładzie tak nie jest, bowiem np. zmiana  $q_3$  powoduje ruch trzeciego i czwartego przegubu manipulatora (zobacz tabela 2.1). W związku z tym, musimy najpierw dokonać zamiany współrzędnych przegubowych, tak by wspomniane założenie było spełnione. Można to osiągnąć przyjmując

$$\mathbf{q} = \boldsymbol{\varphi}(\check{\mathbf{q}}) = (\check{q}_1, \check{q}_2, \check{q}_3, \check{q}_3 + \check{q}_4, \check{q}_3 + \check{q}_4 + \check{q}_5, \check{q}_6)^\top.$$

Teraz, wykorzystując poszczególne macierze  $\mathbf{A}_0^i(\check{\mathbf{q}})$ ,  $i = 1, \dots, 5$ , oraz zależność (2.98), wyliczamy kolejne kolumny macierzy  $\check{\mathbf{J}}^s(\check{\mathbf{q}})$  wyrażonej w nowych współrzędnych. Po powrocie do współrzędnych oryginalnych  $\mathbf{q}$ , zgodnie z zależnością

$$\mathbf{J}^s(\mathbf{q}) = \check{\mathbf{J}}^s(\boldsymbol{\varphi}^{-1}(\mathbf{q})) \frac{\partial \boldsymbol{\varphi}^{-1}}{\partial \mathbf{q}}(\mathbf{q}),$$

otrzymujemy macierz

$$\mathbf{J}^s(\mathbf{q}) = \begin{bmatrix} 0 & q_1 & a_2 c_2 s_3 & a_3 c_2 s_4 & c_2(d_1 + a_2 s_3 + a_3 s_4) & \\ 0 & 0 & -a_2 c_3 & -a_3 c_4 & q_1 s_2 + a_2 c_3 + a_3 c_4 & \\ 1 & 0 & -a_2 s_2 s_3 & -a_3 s_2 s_4 & s_2(d_1 + a_2 s_3 + a_3 s_4) & \\ 0 & 0 & 0 & 0 & s_2 & \\ 0 & -1 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & -c_2 & \\ & & & & & -q_1 c_5 - s_2(d_1 s_5 + a_2 c_{3-5} + a_3 c_{4-5}) \\ & & & & & q_1 c_2 s_5 \\ & & & & & c_2(d_1 s_5 + a_2 c_{3-5} + a_3 c_{4-5}) \\ & & & & & c_2 s_5 \\ & & & & & c_5 \\ & & & & & s_2 s_5 \end{bmatrix}, \quad (2.105)$$

w której  $c_{i-j}$  oznacza  $\cos(q_i - q_j)$ . ■

Zauważmy, że wyprowadzona przez nas postać ogólna jacobianu geometrycznego w przestrzeni różni się od postaci tzw. *jacobianu manipulatora*

podawanej w podręcznikach robotyki. Przyczyna różnicy tkwi w definicji tego obiektu. Mianowicie, jacobian manipulatora jest rozumiany jako macierz przekształcenia prędkości ruchu w przegubach w prędkość liniową i prędkość kątową w przestrzeni efektora względem układu podstawowego

$$\begin{pmatrix} \dot{\mathbf{T}}_0^n \\ \boldsymbol{\omega}_s \end{pmatrix} = \mathbf{J}^m(\mathbf{q})\dot{\mathbf{q}}. \quad (2.106)$$

Dla  $i$ -tego przegubu obrotowego otrzymujemy przy takiej definicji jacobianu

$$\mathbf{J}_i^m(\mathbf{q}) = \begin{pmatrix} \mathbf{R}_{03\text{kol}}^{i-1}(\mathbf{q}^{i-1}) \times (\mathbf{T}_0^n(\mathbf{q}) - \mathbf{T}_0^{i-1}(\mathbf{q}^{i-1})) \\ \mathbf{R}_{03\text{kol}}^{i-1}(\mathbf{q}^{i-1}) \end{pmatrix}, \quad (2.107)$$

gdzie  $\mathbf{q}^{i-1} = (q_1, \dots, q_{i-1})^T$ . Kolumny jacobianu  $\mathbf{J}^m(\mathbf{q})$  odpowiadające przegubom przesuwным manipulatora są takie same, jak w przypadku jacobianu  $\mathbf{J}^s(\mathbf{q})$ . Korzystając z formuł (2.98) oraz (2.106) nietrudno pokazać, że istnieje następujący związek między jacobianem geometrycznym w przestrzeni a jacobianem manipulatora

$$\mathbf{J}^m(\mathbf{q}) = \begin{bmatrix} \mathbb{I}_3 & -[\mathbf{T}_0^n(\mathbf{q})] \\ \mathbf{0} & \mathbb{I}_3 \end{bmatrix} \mathbf{J}^s(\mathbf{q}). \quad (2.108)$$

Przypominamy, że odwzorowanie [ ] zostało zdefiniowane formułą (2.10).

### Przykład 2.3.20 (Manipulator typu podwójne wahadło)

Postępując analogicznie jak w przykładzie 2.3.16, na podstawie zależności (2.107), możemy wyznaczyć jacobian manipulatora typu podwójne wahadło

$$\mathbf{J}^m(\mathbf{q}) = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} & -l_2 s_{12} \\ l_1 c_1 + l_2 c_{12} & l_2 c_{12} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}. \quad (2.109)$$

■



**Przykład 2.3.21 (Manipulator typu potrójne wahadło)**

Wykorzystanie zależności (2.107) pozwala otrzymać jakobian manipulatora typu potrójne wahadło

$$\mathbf{J}^m(\mathbf{q}) = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} - l_3 s_{123} & -l_2 s_{12} - l_3 s_{123} & -l_3 s_{123} \\ l_1 c_1 + l_2 c_{12} + l_3 c_{123} & l_2 c_{12} + l_3 c_{123} & l_3 c_{123} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}. \quad (2.110)$$

■

**Przykład 2.3.22 (Manipulator typu SCARA)**

Na podstawie zależności (2.107) i (2.99) otrzymujemy jakobian manipulatora typu SCARA<sup>§</sup>

$$\mathbf{J}^m(\mathbf{q}) = \begin{bmatrix} -a_1 s_1 - a_2 s_{12} & -a_2 s_{12} & 0 & 0 \\ a_1 c_1 + a_2 c_{12} & a_2 c_{12} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}. \quad (2.111)$$

■

W celu wyznaczenia jakobianu geometrycznego w ciele, wystarczy skorzystać z następującej relacji między prędkościami w ciele i w przestrzeni

$$\begin{pmatrix} \mathbf{v}_b \\ \boldsymbol{\omega}_b \end{pmatrix} = \begin{bmatrix} \mathbf{R}^T & -\mathbf{R}^T[\mathbf{T}] \\ \mathbf{0} & \mathbf{R}^T \end{bmatrix} \begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix},$$

łatwej do wyprowadzenia na podstawie wzorów (2.6), (2.9), (2.12). Zakładając, że  $\mathbf{R}(\mathbf{q})$ ,  $\mathbf{T}(\mathbf{q})$  są określone wyrażeniem (2.42), otrzymujemy

$$\mathbf{J}^b(\mathbf{q}) = \begin{bmatrix} \mathbf{R}_0^{nT}(\mathbf{q}) & -\mathbf{R}_0^{nT}(\mathbf{q})[\mathbf{T}_0^n(\mathbf{q})] \\ \mathbf{0} & \mathbf{R}_0^{nT}(\mathbf{q}) \end{bmatrix} \mathbf{J}^s(\mathbf{q}). \quad (2.112)$$

Przypomnijmy, że dla wektora  $\mathbf{T} = (T_1, T_2, T_3)^T$ ,  $[\mathbf{T}]$  oznacza macierz skośnie symetryczną (2.10).

<sup>§</sup>Do wycalczenia kolumn macierzy użyto tych samych macierzy, co w przypadku jakobianu  $\mathbf{J}^s$ .

W postaci jawnej jacobian geometryczny w ciele wygląda następująco. Jeżeli przegub o numerze  $i$  jest obrotowy, to  $i$ -ta kolumna macierzy  $\mathbf{J}^b(\mathbf{q})$  wyraża się jako

$$\mathbf{J}_i^b(\mathbf{q}) = \begin{pmatrix} (\mathbf{R}_{i-1}^{nT})_{3\text{kol}}(\mathbf{q}_i) \times \mathbf{R}_{i-1}^{nT}(\mathbf{q}_i) \mathbf{T}_{i-1}^n(\mathbf{q}_i) \\ (\mathbf{R}_{i-1}^{nT})_{3\text{kol}}(\mathbf{q}_i) \end{pmatrix}, \quad (2.113)$$

gdzie  $\mathbf{q}_i = (q_i, \dots, q_n)^T$ . Gdy  $i$ -ty przegub jest przesuwny, to

$$\mathbf{J}_i^b(\mathbf{q}) = \begin{pmatrix} (\mathbf{R}_{i-1}^{nT})_{3\text{kol}}(\mathbf{q}_i) \\ \mathbf{0} \end{pmatrix}. \quad (2.114)$$

We wzorach (2.113), (2.114) symbolem  $(\mathbf{R}_{i-1}^{nT})_{3\text{kol}}$  oznaczyliśmy trzecią kolumnę macierzy  $\mathbf{R}_{i-1}^{nT}(\mathbf{q}_i)$  zdefiniowanej, tak samo jak wektor  $\mathbf{T}_{i-1}^n(\mathbf{q}_i)$ , za pośrednictwem formuły

$$\begin{bmatrix} \mathbf{R}_{i-1}^n(\mathbf{q}_i) & \mathbf{T}_{i-1}^n(\mathbf{q}_i) \\ \mathbf{0} & 1 \end{bmatrix} = \prod_{k=i}^n \mathbf{A}_{k-1}^k(q_k).$$

### Przykład 2.3.23 (Manipulator typu podwójne wahadło)

Jacobian geometryczny w ciele manipulatora typu podwójne wahadło, wyliczony na podstawie zależności (2.113), przyjmuje następującą postać<sup>¶</sup>

$$\mathbf{J}^b(\mathbf{q}) = \begin{bmatrix} l_1 s_2 & 0 \\ l_2 + l_1 c_2 & l_2 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}. \quad (2.115)$$

Pozostawiamy Czytelnikowi przyjemność sprawdzenia, że jacobiany geometryczne (2.102) i (2.115) manipulatora typu podwójne wahadło spełniają zależność (2.112). ■

<sup>¶</sup>Pierwszą kolumnę wyliczamy wykorzystując macierz kinematyki manipulatora  $\mathbf{K}$  daną wzorem (2.46), drugą — na podstawie macierzy  $\mathbf{A}_1^2$ .

**Przykład 2.3.24 (Manipulator typu SCARA)**

Jakobian geometryczny w ciele manipulatora typu SCARA znajdujemy na podstawie zależności (2.113) i (2.114)<sup>||</sup>

$$\mathbf{J}^b(\mathbf{q}) = \begin{bmatrix} a_2 s_4 + a_1 s_{24} & a_2 s_4 & 0 & 0 \\ a_2 c_4 + a_1 c_{24} & a_2 c_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}. \quad (2.116)$$

■

Związek pomiędzy jakobianem analitycznym a jakobianem geometrycznym wynika z relacji, jaka zachodzi między prędkością zmian współrzędnych zadaniowych a prędkością efektora w przestrzeni lub w ciele. Załóżmy, dla przykładu, że współrzędne zadaniowe mają postać  $\mathbf{y} = (\text{współrzędne kartezjańskie położenia } \mathbf{T}, \text{ kąty Eulera Z-Y-Z } (\varphi, \theta, \psi))$ . Związek między prędkością w przestrzeni  $\boldsymbol{\omega}_s$  a pochodnymi kątów Eulera Z-Y-Z względem czasu uzyskuje się ze wzoru (2.24) w następujący sposób

$$\begin{aligned} \boldsymbol{\Omega}_s &= \dot{\mathbf{R}}\mathbf{R}^T = \dot{\mathbf{R}}(Z, \varphi)\mathbf{R}^T(Z, \varphi) + \mathbf{R}(Z, \varphi)\dot{\mathbf{R}}(\gamma, \theta)\mathbf{R}^T(\gamma, \theta)\mathbf{R}^T(Z, \varphi) + \\ &+ \mathbf{R}(Z, \varphi)\mathbf{R}(\gamma, \theta)\dot{\mathbf{R}}(Z, \psi)\mathbf{R}^T(Z, \psi)\mathbf{R}^T(\gamma, \theta)\mathbf{R}^T(Z, \varphi) = \\ &= [\mathbf{e}_3]\dot{\varphi} + [\mathbf{R}(Z, \varphi)\mathbf{e}_2]\dot{\theta} + [\mathbf{R}(Z, \varphi)\mathbf{R}(\gamma, \theta)\mathbf{e}_3]\dot{\psi}. \end{aligned}$$

Przechodząc do prędkości wektorowej w przestrzeni uzyskujemy zależność

$$\boldsymbol{\omega}_s = \mathbf{E}(\varphi, \theta, \psi) \begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix}, \quad (2.117)$$

gdzie

$$\begin{aligned} \mathbf{E}(\varphi, \theta, \psi) &= [\mathbf{e}_3, \mathbf{R}(Z, \varphi)\mathbf{e}_2, \mathbf{R}(Z, \varphi)\mathbf{R}(\gamma, \theta)\mathbf{e}_3] = \\ &= \begin{bmatrix} 0 & -\sin \varphi & \cos \varphi \sin \theta \\ 0 & \cos \varphi & \sin \varphi \sin \theta \\ 1 & 0 & \cos \theta \end{bmatrix}. \end{aligned}$$

<sup>||</sup>Poszczególne kolumny tego jakobianu wylicza się, odpowiednio, na podstawie macierzy  $\mathbf{A}_0^4 = \mathbf{K}, \mathbf{A}_1^4, \mathbf{A}_2^4, \mathbf{A}_3^4$ .

Dla współrzędnych kartezjańskich położenia wyliczamy

$$\mathbf{v}_s = \dot{\mathbf{T}} - \boldsymbol{\Omega}_s \mathbf{T} = \dot{\mathbf{T}} + [\mathbf{T}] \boldsymbol{\omega}_s.$$

W rezultacie, transformacja prędkości przyjmuje następującą postać

$$\begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix} = \begin{bmatrix} \mathbb{I}_3 & [\mathbf{T}] \mathbf{E}(\varphi, \theta, \psi) \\ \mathbf{0} & \mathbf{E}(\varphi, \theta, \psi) \end{bmatrix} \begin{pmatrix} \dot{T}_1 \\ \dot{T}_2 \\ \dot{T}_3 \\ \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix}. \quad (2.118)$$

Po utożsamieniu współrzędnych przegubowych z położeniami przegubów manipulatora ( $\mathbf{x} = \mathbf{q}$ ) znajdujemy następujący związek między jakobianem analitycznym  $\mathbf{J}^a(\mathbf{x})$  odnoszącym się do współrzędnych (współrzędne kartezjańskie, kąty Eulera Z-Y-Z) w specjalnej grupie euklidesowej a jakobianem geometrycznym  $\mathbf{J}^s(\mathbf{x})$  w przestrzeni

$$\mathbf{J}^s(\mathbf{x}) = \begin{bmatrix} \mathbb{I}_3 & [\mathbf{T}] \mathbf{E}(\varphi, \theta, \psi) \\ \mathbf{0} & \mathbf{E}(\varphi, \theta, \psi) \end{bmatrix} \mathbf{J}^a(\mathbf{x}). \quad (2.119)$$

Podobne relacje można wyprowadzić dla innych układów współrzędnych (parametryzacji) grupy  $\mathbb{SE}(3)$  i jej podgrup. W przypadku parametryzacji kołysanie-kiwanie-myszkowanie, prędkość kątowa w przestrzeni wynosi

$$\boldsymbol{\omega}_s = \mathbf{KKM}(\varphi, \theta, \psi) \begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{bmatrix} 0 & -\sin \varphi & \cos \varphi \cos \theta \\ 0 & \cos \varphi & \sin \varphi \cos \theta \\ 1 & 0 & -\sin \theta \end{bmatrix} \begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix}. \quad (2.120)$$

Po wzięciu pod uwagę definicji prędkości liniowej w przestrzeni oraz zależności (2.120) otrzymujemy następującą transformację prędkości

$$\begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix} = \begin{bmatrix} \mathbb{I}_3 & [\mathbf{T}] \mathbf{KKM}(\varphi, \theta, \psi) \\ \mathbf{0} & \mathbf{KKM}(\varphi, \theta, \psi) \end{bmatrix} \begin{pmatrix} \dot{T}_1 \\ \dot{T}_2 \\ \dot{T}_3 \\ \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix}. \quad (2.121)$$

Dla kompletności dodajmy transformację prędkości związaną z parametryzacją Cayleya. Dla prędkości kątowej w przestrzeni otrzymujemy związek

$$\boldsymbol{\omega}_s = \mathbf{C}(\mathbf{r})\dot{\mathbf{r}}, \quad (2.122)$$

w którym macierz  $\mathbf{C}(\mathbf{r})$  wyraża się formułą

$$\mathbf{C}(\mathbf{r}) = 2 \begin{bmatrix} 1 + r_1^2 & r_1 r_2 + r_3 & r_1 r_3 - r_2 \\ r_1 r_2 - r_3 & 1 + r_2^2 & r_2 r_3 + r_1 \\ r_1 r_3 + r_2 & r_2 r_3 - r_1 & 1 + r_3^2 \end{bmatrix}^{-1}. \quad (2.123)$$

W rezultacie, transformacja prędkości w przypadku parametryzacji Cayleya posiada następującą reprezentację macierzową

$$\begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix} = \begin{bmatrix} \mathbb{I}_3 & [\mathbf{T}]\mathbf{C}(\mathbf{r}) \\ \mathbf{0} & \mathbf{C}(\mathbf{r}) \end{bmatrix} \begin{pmatrix} \dot{T}_1 \\ \dot{T}_2 \\ \dot{T}_3 \\ \dot{r}_1 \\ \dot{r}_2 \\ \dot{r}_3 \end{pmatrix}. \quad (2.124)$$

### Przykład 2.3.25 (Manipulator typu podwójne wahadło)

W tym przykładzie zajmiemy się wyprowadzeniem zależności analogicznej do (2.119), wiążącej jacobian analityczny  $\mathbf{J}^a$  manipulatora typu podwójne wahadło, zdefiniowanego wzorem (2.81), z jego jacobianem geometrycznym w przestrzeni  $\mathbf{J}^s$ , określonym przez (2.102). Przed przystąpieniem do tego zauważmy, że jacobiany geometryczne (2.102) i (2.115) wyliczono z ogólnych zależności zakładających, że ruch odbywa się w grupie  $\mathbb{SE}(3)$ , natomiast jacobian analityczny (2.81) został wyliczony dla kinematyki wyrażonej we współrzędnych reprezentujących grupę  $\mathbb{SE}(2)$ . Uwzględnienie tego faktu spowoduje zmianę rozmiarów macierzy określającej związek między jacobianami w stosunku do macierzy (2.119).

Współrzędne zadaniowe  $\mathbf{y}$  wybrane w grupie  $\mathbb{SE}(2)$  odpowiadają w grupie  $\mathbb{SE}(3)$  położeniu wzdłuż osi  $X$  i  $Y$  podstawowego układu współrzędnych  $X_0 Y_0 Z_0$  oraz obrotowi wokół osi  $Z$  tego układu. Dlatego, dla tak wybranych współrzędnych

$$\mathbf{T}(\mathbf{y}) = \begin{pmatrix} y_1 \\ y_2 \\ 0 \end{pmatrix}, \quad \mathbf{R}(\mathbf{y}) = \mathbf{R}(Z, y_3) = \begin{bmatrix} \cos y_3 & -\sin y_3 & 0 \\ \sin y_3 & \cos y_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Jak pokazaliśmy, macierz prędkości kątovej w przestrzeni  $\boldsymbol{\Omega}_s = \dot{\mathbf{R}}\mathbf{R}^T = [\boldsymbol{\omega}_s]$  jest całkowicie określona przez wektor prędkości kątovej w przestrzeni  $\boldsymbol{\omega}_s = (\omega_{s1}, \omega_{s2}, \omega_{s3})^T$ . W naszym przypadku

$$\boldsymbol{\Omega}_s = \begin{bmatrix} 0 & -\dot{y}_3 & 0 \\ \dot{y}_3 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

co oznacza, że  $\boldsymbol{\omega}_s = \mathbf{e}_3\dot{y}_3$ . Wiemy również, że wektor prędkości liniowej w przestrzeni jest równy  $\mathbf{v}_s = \dot{\mathbf{T}} - \boldsymbol{\Omega}_s\mathbf{T}$ , co w analizowanym przykładzie sprowadza się do

$$\mathbf{v}_s = \begin{pmatrix} \dot{y}_1 + y_2\dot{y}_3 \\ \dot{y}_2 - y_1\dot{y}_3 \\ 0 \end{pmatrix} = \begin{bmatrix} 1 & 0 & y_2 \\ 0 & 1 & -y_1 \\ 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{pmatrix}.$$

Zestawiając otrzymane wyrażenia na  $\boldsymbol{\omega}_s$  i  $\mathbf{v}_s$  uzyskujemy

$$\begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix} = \begin{bmatrix} 1 & 0 & y_2 \\ 0 & 1 & -y_1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{pmatrix} = \mathbf{F}(\mathbf{y})\dot{\mathbf{y}}.$$

Po naturalnym utożsamieniu współrzędnych konfiguracyjnych  $\mathbf{q}$  z położeniami przegubów manipulatora  $\mathbf{x}$ , określamy związek między jakobianem analitycznym  $\mathbf{J}^a(\mathbf{x})$  odnoszącym się do wybranego układu współrzędnych a jakobianem geometrycznym  $\mathbf{J}^s(\mathbf{x})$  w przestrzeni

$$\mathbf{J}^s(\mathbf{x}) = \mathbf{F}(\mathbf{k}(\mathbf{x}))\mathbf{J}^a(\mathbf{x}), \quad (2.125)$$

gdzie  $\mathbf{k}(\mathbf{x})$  oznacza kinematykę podwójnego wahadła wyrażoną we współrzędnych w postaci (2.67). Łatwo sprawdzić, że wyliczone wcześniej jakobiany (2.81) i (2.102) spełniają zależność (2.125).

Zależności typu (2.125) są ważne z obliczeniowego punktu widzenia. Pozwalają one bowiem wyliczyć jakobian analityczny w sytuacjach, gdy nie jesteśmy w stanie wyrazić kinematyki manipulatora we współrzędnych w sposób jawny. ■

**Przykład 2.3.26 (Manipulator typu SCARA)**

W celu wyznaczenia relacji wiążącej jacobian analityczny  $\mathbf{J}^a$  manipulatora SCARA z jego jacobianem geometrycznym w przestrzeni  $\mathbf{J}^s$  należy postąpić dokładnie w taki sam sposób, jak w przypadku manipulatora typu podwójne wahadło (przykład 2.3.25). Po wykonaniu niezbędnych przekształceń otrzymujemy związek postaci

$$\mathbf{J}^s(\mathbf{x}) = \begin{bmatrix} 1 & 0 & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & 1 & 0 & -a_1 c_1 - a_2 c_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{J}^a(\mathbf{x}). \quad (2.126)$$

■

**Przykład 2.3.27 (Manipulator IRb-6 na torze jezdny)**

Na zakończenie znajdziemy zależność wiążącą jacobian analityczny manipulatora IRb-6 zamontowanego na torze jezdny, zdefiniowany przez (2.87), z jego jacobianem geometrycznym w przestrzeni określonym przez (2.105). Dla wybranych współrzędnych kinematyki\*\* otrzymujemy

$$\begin{aligned} \mathbf{T}(\mathbf{y}) &= (y_1, y_2, y_3)^T, \\ \mathbf{R}(\mathbf{y}) &= \mathbf{R}(Y, y_4)\mathbf{R}(Z, y_5)\mathbf{R}(Y, y_6) = \\ &= \begin{bmatrix} \check{c}_4\check{c}_5\check{c}_6 - \check{s}_4\check{s}_6 & -\check{c}_4\check{s}_5 & \check{s}_4\check{c}_6 + \check{c}_4\check{c}_5\check{s}_6 \\ \check{s}_5\check{c}_6 & \check{c}_5 & \check{s}_5\check{s}_6 \\ -\check{s}_4\check{c}_5\check{c}_6 - \check{c}_4\check{s}_6 & \check{s}_4\check{s}_5 & \check{c}_4\check{c}_6 - \check{s}_4\check{c}_5\check{s}_6 \end{bmatrix}, \end{aligned}$$

gdzie  $\check{s}_i$  i  $\check{c}_j$  oznaczają  $\sin y_i$  i  $\cos y_i$ . Wyliczenie macierzy  $\mathbf{\Omega}_s = \dot{\mathbf{R}}\mathbf{R}^T$  pozwala na określenie wektora prędkości kątowej

$$\boldsymbol{\omega}_s = \begin{pmatrix} \check{s}_4\dot{y}_5 - \check{c}_4\check{s}_5\dot{y}_6 \\ \dot{y}_4\check{c}_5\dot{y}_6 \\ \check{c}_4\dot{y}_5 + \check{s}_4\check{s}_5\dot{y}_6 \end{pmatrix} = \begin{bmatrix} 0 & \check{s}_4 & -\check{c}_4\check{s}_5 \\ 1 & 0 & \check{c}_5 \\ 0 & \check{c}_4 & \check{s}_4\check{s}_5 \end{bmatrix} \begin{pmatrix} \dot{y}_4 \\ \dot{y}_5 \\ \dot{y}_6 \end{pmatrix}.$$

Następnie, po znalezieniu wektora  $\mathbf{v}_s = \dot{\mathbf{T}} - \mathbf{\Omega}_s\mathbf{T}$  postaci

$$\mathbf{v}_s = \begin{pmatrix} \dot{y}_1 - y_3\dot{y}_4 + y_2\check{c}_4\dot{y}_5 + (y_2\check{s}_4\check{s}_5 - y_3\check{c}_5)\dot{y}_6 \\ \dot{y}_2 + (y_3\check{s}_4 - y_1\check{c}_4)\dot{y}_5 - (y_1\check{s}_4\check{s}_5 + y_3\check{c}_4\check{s}_5)\dot{y}_6 \\ \dot{y}_3 + y_1\dot{y}_4 - y_2\check{s}_4\dot{y}_5 + (y_1\check{c}_5 + y_2\check{c}_4\check{s}_5)\dot{y}_6 \end{pmatrix},$$

---

\*\*Są to współrzędne kartezjańskie położenia efektora wraz z kątami Eulera Y-Z-Y określającymi jego orientację.

otrzymujemy następujący związek między prędkościami w przestrzeni a prędkościami we współrzędnych zadaniowych

$$\begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix} = \begin{bmatrix} 1 & 0 & 0 & -y_3 & y_2\check{c}_4 & y_2\check{s}_4\check{s}_5 - y_3\check{c}_5 \\ 0 & 1 & 0 & 0 & -y_1\check{c}_4 + y_3\check{s}_4 & y_1\check{s}_4\check{s}_5 + y_3\check{c}_4\check{s}_5 \\ 0 & 0 & 1 & y_1 & -y_2\check{s}_4 & y_1\check{c}_5 + y_2\check{c}_4\check{s}_5 \\ 0 & 0 & 0 & 0 & \check{s}_4 & -\check{c}_4\check{s}_5 \\ 0 & 0 & 0 & 1 & 0 & \check{c}_5 \\ 0 & 0 & 0 & 0 & \check{c}_4 & \check{s}_4\check{s}_5 \end{bmatrix} \begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \\ \dot{y}_4 \\ \dot{y}_5 \\ \dot{y}_6 \end{pmatrix}. \quad (2.127)$$

Na podstawie formuły (2.127), po utożsamieniu współrzędnych konfiguracyjnych  $\mathbf{q}$  z położeniami przegubów manipulatora  $\mathbf{x}$ , otrzymujemy poszukiwany związek między jacobianem analitycznym  $\mathbf{J}^a(\mathbf{x})$  odnoszącym się do wybranych współrzędnych zadaniowych a jacobianem geometrycznym  $\mathbf{J}^s(\mathbf{x})$  w przestrzeni

$$\mathbf{J}^s(\mathbf{x}) = \begin{bmatrix} 1 & 0 & 0 & -x_1 - s_2 a_{23} & c_2(d_1 + a_2 s_3 + a_3 s_4 - d_6 c_5) \\ 0 & 1 & 0 & 0 & -x_1 s_2 - a_2 c_3 - a_3 c_4 - d_6 s_5 \\ 0 & 0 & 1 & c_2 a_{23} & -s_2(d_1 + a_2 s_3 + a_3 s_4 - d_6 c_5) \\ 0 & 0 & 0 & 0 & -s_2 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & c_2 \end{bmatrix} \mathbf{J}^a(\mathbf{x}), \quad (2.128)$$

$$\begin{bmatrix} x_1 c_5 + s_2(d_1 s_5 + a_2 c_{3-5} + a_3 c_{4-5}) \\ -x_1 c_2 s_5 \\ -c_2(d_1 s_5 + a_2 c_{3-5} + a_3 c_{4-5}) \\ -c_2 s_5 \\ -c_5 \\ -s_2 s_5 \end{bmatrix}$$

przy oznaczeniach  $a_{23} = a_2 c_3 + a_3 c_4 + d_6 s_5$ ,  $c_{i-j} = \cos(x_i - x_j)$ . ■

Kończąc rozważania na temat kinematyki, zwróćmy uwagę na zalety wykładniczej reprezentacji kinematyki manipulatora ujawniające się przy obliczaniu jacobianu geometrycznego manipulatora w przestrzeni. Mianowicie, nietrudno pokazać, że formuła prędkości efektora w przestrzeni przyjmuje postać

$$\begin{aligned} \mathbf{V}^s &= \dot{\mathbf{K}}(\mathbf{q})\mathbf{K}^{-1}(\mathbf{q}) = \\ &= \zeta_1 \dot{q}_1 + \exp(\zeta_1 q_1) \zeta_2 \exp(-\zeta_1 q_1) \dot{q}_2 + \dots + \exp(\zeta_1 q_1) \dots \\ &\quad \dots \exp(\zeta_{n-1} q_{n-1}) \zeta_n \exp(-\zeta_{n-1} q_{n-1}) \dots \exp(-\zeta_1 q_1) \dot{q}_n. \quad (2.129) \end{aligned}$$



Dzięki zależności (2.58)

$$\exp(\zeta_i q_i) = \begin{bmatrix} \exp([\boldsymbol{\omega}_i]q_i) & -(\boldsymbol{\omega}_i \times \mathbf{p}_i)q_i \\ \mathbf{0} & 1 \end{bmatrix}. \quad (2.130)$$

Macierze występujące we wzorze (2.129) po lewej stronie prędkości ruchu przegubów należą do algebry Liego  $\mathfrak{se}(3)$  grupy  $\mathbb{SE}(3)$  i są zdefiniowane przez wektory z przestrzeni  $\mathbb{R}^6$  (skrętniki), które można potraktować bezpośrednio jako kolumny jacobianu geometrycznego w przestrzeni.

### Przykład 2.3.28 (Manipulator typu podwójne wahadło)

Zgodnie z formułą (2.130), korzystając z reprezentacji wykładniczej kinematyki (2.60), (2.61) możemy obliczyć prędkość w przestrzeni efektora manipulatora typu podwójne wahadło

$$\mathbf{V}^s = \dot{\mathbf{K}}(\mathbf{q})\mathbf{K}^{-1}(\mathbf{q}) = \zeta_1 \dot{q}_1 + \exp(\zeta_1 q_1) \zeta_2 \exp(-\zeta_1 q_1) \dot{q}_2. \quad (2.131)$$

Macierze, przez które są mnożone prędkości w (2.131) należą do algebry Liego  $\mathfrak{se}(3)$ , a odpowiadające im skrętniki definiują kolumny jacobianu geometrycznego w przestrzeni. Jak łatwo zauważyć, macierz  $\zeta_1 = \begin{bmatrix} [\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}$  jest równoważna wektorowi  $\mathbf{e}_6 \in \mathbb{R}^6$ . Druga z macierzy

$$\begin{aligned} & \exp(\zeta_1 q_1) \zeta_2 \exp(-\zeta_1 q_1) = \\ & = \begin{bmatrix} \mathbf{R}(Z, q_1) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} [\mathbf{e}_3] & -l_1 \mathbf{e}_2 \\ \mathbf{0} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R}^T(Z, q_1) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} [\mathbf{e}_3] & -l_1 \mathbf{R}(Z, q_1) \mathbf{e}_2 \\ \mathbf{0} & 0 \end{bmatrix}, \end{aligned}$$

ma skrętnik  $(l_1 s_1, -l_1 c_1, 0, 0, 0, 1)^T$ . W rezultacie, jacobian geometryczny w przestrzeni

$$\mathbf{J}^s(\mathbf{q}) = \begin{bmatrix} 0 & l_1 s_1 \\ 0 & -l_1 c_1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}, \quad (2.132)$$

obliczony przy wykorzystaniu reprezentacji wykładniczej kinematyki podwójnego wahadła jest taki sam, jak otrzymany w przykładzie 2.3.16. ■

**Przykład 2.3.29 (Manipulator typu SCARA)**

Korzystając z reprezentacji wykładniczej (2.63), (2.64) kinematyki manipulatora SCARA, obliczamy prędkość efektora w przestrzeni

$$\begin{aligned} \mathbf{V}^s = \mathbf{K}(\mathbf{q})\mathbf{K}^{-1}(\mathbf{q}) &= \boldsymbol{\zeta}_1 \dot{q}_1 + \exp(\boldsymbol{\zeta}_1 q_1) \dot{q}_2 + \\ &+ \exp(\boldsymbol{\zeta}_1 q_1) \exp(\boldsymbol{\zeta}_2 q_2) \boldsymbol{\zeta}_3 \exp(-\boldsymbol{\zeta}_2 q_2) \exp(-\boldsymbol{\zeta}_1 q_1) \dot{q}_3 + \exp(\boldsymbol{\zeta}_1 q_1) \exp(\boldsymbol{\zeta}_2 q_2) \\ &\exp(\boldsymbol{\zeta}_3 q_3) \boldsymbol{\zeta}_4 \exp(-\boldsymbol{\zeta}_3 q_3) \exp(-\boldsymbol{\zeta}_2 q_2) \exp(-\boldsymbol{\zeta}_1 q_1) \dot{q}_4. \end{aligned} \quad (2.133)$$

Jako kolumny jacobianu geometrycznego w przestrzeni manipulatora SCARA weźmiemy skrętniki odpowiadające macierzom, przez które są mnożone prędkości przegubów manipulatora we wzorze (2.133). Po wykonaniu elementarnych operacji algebraicznych otrzymujemy następujące wyniki:

$$\begin{aligned} \boldsymbol{\zeta}_1 &= \begin{bmatrix} [\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} \cong \mathbf{e}_6 \in \mathbb{R}^6, \\ \exp(\boldsymbol{\zeta}_1 q_1) \boldsymbol{\zeta}_2 \exp(-\boldsymbol{\zeta}_1 q_1) &= \begin{bmatrix} [\mathbf{e}_3] & -a_1 \mathbf{R}(Z, q_1) \mathbf{e}_2 \\ \mathbf{0} & 0 \end{bmatrix} \cong \begin{pmatrix} a_1 s_1 \\ -a_1 c_1 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \in \mathbb{R}^6, \end{aligned}$$

$$\begin{aligned} \exp(\boldsymbol{\zeta}_1 q_1) \exp(\boldsymbol{\zeta}_2 q_2) \boldsymbol{\zeta}_3 \exp(-\boldsymbol{\zeta}_2 q_2) \exp(-\boldsymbol{\zeta}_1 q_1) &= \boldsymbol{\zeta}_3 \cong \mathbf{e}_3 \in \mathbb{R}^6, \\ \exp(\boldsymbol{\zeta}_1 q_1) \exp(\boldsymbol{\zeta}_2 q_2) \exp(\boldsymbol{\zeta}_3 q_3) \boldsymbol{\zeta}_4 \exp(-\boldsymbol{\zeta}_3 q_3) \exp(-\boldsymbol{\zeta}_2 q_2) \exp(-\boldsymbol{\zeta}_1 q_1) &= \\ = \begin{bmatrix} [\mathbf{e}_3] & -\mathbf{R}(Z, q_1) \begin{pmatrix} a_2 s_2 \\ -a_1 - a_2 c_2 \\ 0 \end{pmatrix} \\ \mathbf{0} & 0 \end{bmatrix} &\cong \begin{pmatrix} a_1 s_1 + a_2 s_{12} \\ -a_1 c_1 - a_2 c_{12} \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \in \mathbb{R}^6. \end{aligned}$$

W rezultacie, jacobian geometryczny w przestrzeni

$$\mathbf{J}^s(\mathbf{q}) = \begin{bmatrix} 0 & a_1 s_1 & 0 & a_1 s_1 + a_2 s_{12} \\ 0 & -a_1 c_1 & 0 & -a_1 c_1 - a_2 c_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix} \quad (2.134)$$

ma postać identyczną z (2.104). ■

### 2.3.5 Konfiguracje osobliwe

Konfiguracje, jakie przyjmuje manipulator w czasie ruchu dzielą się na dwie klasy: *konfiguracje regularne (nieosobliwe)* i *konfiguracje osobliwe*. Mówimy, że konfiguracja manipulatora jest regularna, jeżeli jacobian geometryczny (w przestrzeni lub w ciele) ma pełny rząd wierszowy, to znaczy rząd jacobianu jest równy wymiarowi rozmierności zadaniowej. Zgodnie z powyższą definicją, w konfiguracji regularnej przekształcenie

$$\mathbf{J}^s(\mathbf{q}) : \dot{\mathbf{q}} \mapsto \begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix}$$

jest *suriekcją liniową*. Konfigurację  $\mathbf{q}$  manipulatora, w której rząd jacobianu geometrycznego jest mniejszy od wymiaru rozmierności zadaniowej,

$$\text{rank } \mathbf{J}^s(\mathbf{q}) < \dim \mathcal{Z},$$

nazywamy konfiguracją osobliwą. Zauważmy, że manipulator, którego liczba stopni swobody jest mniejsza od wymiaru rozmierności zadaniowej, ma wyłącznie konfiguracje osobliwe. Z definicji jacobianu geometrycznego wynika, że w konfiguracji osobliwej istnieją takie wektory prędkości ruchu przegubów, które nie powodują ruchu efektora manipulatora. Oznacza to, że manipulator traci *zręczność* ruchu. Stopień utraty zręczności ruchu w konfiguracji osobliwej mierzy się liczbą zwaną *korzędem* tej konfiguracji

$$\text{corank } \mathbf{q} = \dim \mathcal{Z} - \text{rank } \mathbf{J}^s(\mathbf{q}). \quad (2.135)$$

Korząd konfiguracji osobliwej określa liczbę niezależnych kierunków ruchu w przestrzeni stycznej do rozmierności zadaniowej w punkcie  $\mathbf{K}(\mathbf{q})$ , których nie można wygenerować przy pomocy ruchu przegubów w kierunkach dopuszczalnych w konfiguracji  $\mathbf{q}$ .

Niech będzie dana kinematyka  $\mathbf{K} : \mathcal{Q} \rightarrow \mathcal{Z}$  manipulatora o  $n$  stopniach swobody, wraz z jacobianem geometrycznym  $\mathbf{J}^s(\mathbf{q})$ . Stosownie do podanej definicji, konfiguracje osobliwe manipulatora należą do zbioru

$$\mathcal{S} = \{\mathbf{q} \in \mathcal{Q} \mid \text{rank } \mathbf{J}^s(\mathbf{q}) < \dim \mathcal{Z}\}. \quad (2.136)$$

Utożsamiając konfiguracje ze współrzędnymi przegubowymi możemy zależność (2.136) przepisać w równoważnej postaci

$$\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^n \mid \text{rank } \mathbf{J}^s(\mathbf{x}) < \dim \mathcal{Z}\}.$$

Poświęćmy nieco uwagi zbadaniu podstawowych własności zbioru  $S$ . Jak łatwo zauważyć, wyznaczenie zbioru konfiguracji osobliwych wymaga rozwiązania pewnego układu równań nieliniowych, zatem konfiguracje osobliwe mogą nie być możliwe do wyznaczenia w sposób jawny (symboliczny). Z ciągłości jacobianu geometrycznego jako funkcji konfiguracji wynika, że zbiór  $S$  jest domknięty. Co więcej, ponieważ kinematyka manipulatora jest funkcją analityczną, zbiór  $S$  jest tzw. *zbiorem analitycznym*, to znaczy miejscem zerowym pewnej liczby funkcji analitycznych. Jako zbiór analityczny zbiór konfiguracji osobliwych jest nigdziegęsty w przestrzeni przegubowej (o ile tylko manipulator posiada konfiguracje regularne), a więc jest zbiorem topologicznie małym\*. Generalnie, zbiór  $S$  nie jest podrozmaitością przestrzeni przegubowej, jest jednak sumą podrozmaitości (warstw) ułożonych względem siebie w pewien regularny sposób†. Wymiar zbioru  $S$  definiuje się jako największy wymiar podrozmaitości składających się na ten zbiór. Załóżmy, że wymiar rozmaitości zadaniowej wynosi  $m$ . Niech  $S_k$  oznacza zbiór konfiguracji osobliwych korzędu  $k$ . Jest oczywiste, że

$$S = \bigcup_{k=1}^m S_k.$$

Jeżeli zbiór konfiguracji osobliwych korzędu  $k$  jest podrozmaitością przestrzeni przegubowej, to jego *kwowymiar*

$$\text{codim } S_k = k(n - m + k).$$

Często w celu wyznaczenia konfiguracji osobliwych manipulatora używamy reprezentacji jego kinematyki we współrzędnych oraz jacobianu analitycznego. Uzyskane w ten sposób wyniki będą takie same, jak w przypadku wykorzystywania jacobianu geometrycznego, pod warunkiem że macierze przekształcające jacobian analityczny w geometryczny‡ są nieosobliwe. Osobliwości tych macierzy są źródłem tzw. *osobliwości reprezentacji* kinematyki manipulatora.

### Przykład 2.3.30 (Manipulator typu podwójne wahadło)

Wyznamy zbiór konfiguracji osobliwych dla manipulatora typu podwójne wahadło. Z definicji konfiguracji osobliwej wynika, że ten manipulator,

\*Tzw. zbiorem I kategorii.

†Zwany stratyfikacją Whitneya.

‡Patrz formuła (2.119).

opisany kinematyką  $\mathbf{K}(\mathbf{q})$  postaci (2.46) z jacobianem geometrycznym  $\mathbf{J}^s(\mathbf{q})$  danym zależnością (2.102), ma wyłącznie konfiguracje osobliwe<sup>S</sup>. Podobna sytuacja ma miejsce przy ograniczeniu rozmiatości zadaniowej manipulatora do grupy  $\mathbb{SE}(2)$ . Jej wymiar wciąż jest większy od maksymalnego rzędu jacobianu manipulatora.

Konfiguracje regularne w przestrzeni przegubowej rozpatrywanego manipulatora pojawiają się dopiero po redukcji wymiaru przestrzeni zadaniowej do dwóch. Z taką sytuacją mamy do czynienia, gdy manipulator jest opisany kinematyką (2.68) z odpowiadającym jej jacobianem (2.82), bądź kinematyką (2.69) z jacobianem (2.83). W pierwszym przypadku, wyliczając wyznacznik jacobianu  $\mathbf{J}^a(\mathbf{x})$  otrzymujemy

$$\det \mathbf{J}^a(\mathbf{x}) = l_1 l_2 \sin x_2.$$

Oznacza to, że dla manipulatora typu podwójne wahadło, opisanego kinematyką (2.68), zbiór konfiguracji osobliwych

$$S = \{ \mathbf{x} = (x_1, x_2)^T \in \mathbb{R}^2 \mid \sin x_2 = 0 \}. \quad (2.137)$$

Analogicznie, przy kinematyce (2.69) otrzymujemy zbiór konfiguracji osobliwych

$$S = \{ \mathbf{x} = (x_1, x_2)^T \in \mathbb{R}^2 \mid \sin x_1 = 0 \}. \quad (2.138)$$

■

### Przykład 2.3.31 (Manipulator typu potrójne wahadło)

Podobnie jak w poprzednim przykładzie, manipulator typu potrójne wahadło z kinematyką (2.47) przyjmującą wartości na rozmiatości zadaniowej  $\mathbb{SE}(3)$  jest zawsze osobliwy. Policzenie wyznacznika jacobianu analitycznego  $\mathbf{J}^a(\mathbf{x})$  danego przez (2.84), odpowiadającego kinematyce (2.70), prowadzi do wyznaczenia zbioru konfiguracji osobliwych

$$S = \{ \mathbf{x} = (x_1, x_2, x_3)^T \in \mathbb{R}^3 \mid \sin x_2 = 0 \}. \quad (2.139)$$

W celu znalezienia zbioru konfiguracji osobliwych manipulatora typu potrójne wahadło z kinematyką (2.71), należy określić zestaw

$$\mathbf{D}_2 = \begin{pmatrix} l_1(l_2 s_2 + l_3 s_{23}) \\ l_3(l_2 s_3 + l_1 s_{23}) \\ l_2 l_3 s_3 \end{pmatrix}$$

<sup>S</sup>Wymiar rozmiatości zadaniowej  $\mathbb{SE}(3)$  wynosi 6; rząd macierzy  $\mathbf{J}^s(\mathbf{q})$  — co najwyżej 2.

minorów stopnia 2 jacobianu  $\mathbf{J}^a(\mathbf{x})$  danego zależnością (2.85). Jak widać, wszystkie minory macierzy (2.85) będą równe zero jedynie wtedy, gdy  $s_2 = s_3 = 0$ . Poszukiwany zbiór konfiguracji osobliwych można więc zapisać jako

$$S = \{ \mathbf{x} = (x_1, x_2, x_3)^T \in \mathbb{R}^3 \mid \sin x_2 = \sin x_3 = 0 \}. \quad (2.140)$$

■

### Przykład 2.3.32 (Manipulator typu SCARA)

Ponieważ manipulator typu SCARA opisany kinematyką (2.48) ma mniej niż 6 stopni swobody, przyjmuje on wyłącznie konfiguracje osobliwe. Przy kinematyce (2.72) i ograniczeniach (2.74) konfiguracje osobliwe manipulatora tworzą zbiór

$$S = \{ \mathbf{x} = (x_1, x_2, x_3, x_4)^T \in \mathcal{X} \mid \sin x_2 = 0 \}, \quad (2.141)$$

co wynika z postaci wyznacznika jacobianu analitycznego (2.86). Wyliczenie minorów macierzy (2.86), z usuniętym ostatnim wierszem, prowadzi do wniosku, że zbiór konfiguracji manipulatora SCARA opisanego kinematyką (2.73) jest taki sam jak (2.141). ■

### Przykład 2.3.33 (Manipulator IRb-6 na torze jezdny)

Manipulator IRb-6 zamontowany na torze jezdny z kinematyką zdefiniowaną wzorem (2.49) jest jedynym wśród analizowanych przez nas przykładów manipulatorów o rozmaitości zadaniowej  $\mathbb{SE}(3)$ , którego rozmaitość przegubowa nie składa się wyłącznie z konfiguracji osobliwych. Wyznacznik jacobianu geometrycznego w przestrzeni  $\mathbf{J}^s(\mathbf{q})$ , danego wzorem (2.105), jest równy<sup>¶</sup>

$$\det \mathbf{J}^s(\mathbf{x}) = -a_2 a_3 c_2 s_3 - 4s_5.$$

Biorąc pod uwagę ograniczenia ruchu w przegubach manipulatora wynikające z jego konstrukcji mechanicznej (2.78), zbiór konfiguracji osobliwych możemy zapisać jako

$$S^* = \{ \mathbf{x} = (x_1, \dots, x_6)^T \in \mathcal{X} \mid x_2 = \pm \frac{\pi}{2} \vee x_5 = 0 \vee x_5 = \pi \}. \quad (2.142)$$

Wyznacznik jacobianu  $\mathbf{J}^a(\mathbf{x})$  postaci (2.87) wynosi

$$\det \mathbf{J}^a(\mathbf{x}) = -a_2 a_3 c_2 s_3 - 4. \quad (2.143)$$

<sup>¶</sup>Po utożsamieniu zmiennych  $\mathbf{q}$  ze zmiennymi  $\mathbf{x}$ .

Prowadzi to, ponownie po uwzględnieniu ograniczeń konstrukcyjnych manipulatora, do zdefiniowania zbioru konfiguracji osobliwych formułą

$$S = \{\mathbf{x} = (x_1, \dots, x_6)^T \in \mathcal{X} \mid x_2 = \pm \frac{\pi}{2}\}. \quad (2.144)$$

Inkluzja  $S \subset S^*$ , świadczy o tym, że  $S^*$  zawiera nie tylko osobliwości manipulatora, lecz także osobliwości reprezentacji.

Wyznamy jeszcze osobliwości manipulatora IRb-6 zamontowanego na torze jezdny, traktowanego jako manipulator redundantny z kinematyką (2.77). Po obliczeniu minorów rzędu trzeciego jacobianu analitycznego tego manipulatora stwierdzamy, że znajduje się wśród nich minor postaci  $a_2 a_3 s_{3-4} (a_2 c_3 + a_3 c_4 + d_6 s_5)$ , który w całej przestrzeni przegubowej manipulatora  $\mathcal{X}$  jest różny od zera. Oznacza to, że redundantny manipulator IRb-6 zamontowany na torze jezdny nie ma w ogóle konfiguracji osobliwych. ■

## 2.4 Kinematyka robota mobilnego

Po zdefiniowaniu kinematyki manipulatora, będącego układem holonomicznym, zajmiemy się teraz kinematyką układu robotycznego opisanego w uniwersum fazowym  $\mathbb{R}^N \times \mathbb{R}^N$  za pośrednictwem  $l$  niezależnych ograniczeń nieholonomicznych. Założymy, że układ nie podlega ograniczeniom konfiguracyjnym ( $k = 0$ , a więc liczba stopni swobody  $n = N$ ). Tak określony układ robotyczny będziemy nazywać *nieholonomicznym robotem mobilnym* lub, po prostu, *robotem mobilnym*\*. Zadanie, jakie ma realizować robot mobilny, polega na poruszaniu się w pewnym otoczeniu. Z formalnego punktu widzenia, kinematyka robota mobilnego jest scharakteryzowana przez układ sterowania, liniowy z uwagi na sterowanie, postaci (2.33)

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{q})u_i, \quad (2.145)$$

w którym  $\mathbf{q} \in \mathbb{R}^n$ ,  $\mathbf{u} \in \mathbb{R}^m$ ,  $m = n - l$ . Pola wektorowe sterujące  $\mathbf{g}_1(\mathbf{q}), \dots, \mathbf{g}_m(\mathbf{q})$  są gładkie (a nawet analityczne). Będziemy zakładać, że sterowania układu (2.145) należą do *przestrzeni Hilberta*  $\mathbb{L}_m^2[0, T]$  funkcji całkownych z kwadratem, określonych na pewnym przedziale czasu  $[0, T]$

\*W tej książce terminy: robot nieholonomiczny i robot mobilny są traktowane jako synonimy. Nie znaczy to, że negujemy istnienie holonomicznych robotów mobilnych.

i przyjmujących wartości w  $\mathbb{R}^m$  (praktycznie, często będziemy mieć na myśli sterowania odcinkami ciągłe lub odcinkami stałe, o skończonej liczbie punktów nieciągłości).

Załóżmy, że dla każdego stanu początkowego  $\mathbf{q}_0$  i zadanego sterowania  $\mathbf{u}(\cdot)$  istnieje trajektoria  $\mathbf{q}(t)$  układu (2.145) określona na przedziale  $[0, T]$ . Zdefiniujmy odwzorowanie przejścia stanów układu (2.145)

$$\mathbf{q}(t) = \boldsymbol{\varphi}_t(\mathbf{q}_0, \mathbf{u}(\cdot)), \quad (2.146)$$

które stanowi początkowemu układowi  $\mathbf{q}_0$  i sterowaniu  $\mathbf{u}(\cdot)$  przyporządkowuje stan układu w chwili  $t \in [0, T]$ . Oczywiście, dla  $t = 0$  mamy zawsze  $\boldsymbol{\varphi}_0(\mathbf{q}_0, \mathbf{u}(\cdot)) = \mathbf{q}_0$ . Przy ustalonym stanie początkowym  $\mathbf{q}_0$  i zmieniającym się sterowaniu  $\mathbf{u}(\cdot)$  odwzorowanie (2.146) opisuje stany osiągalne w układzie sterowania (2.145) ze stanu  $\mathbf{q}_0$  w chwili  $t$ . Jeżeli dla każdego  $\mathbf{q}_0$  zbiór stanów osiągalnych w chwili  $t$  obejmuje całą przestrzeń stanu  $\mathbb{R}^n$ , układ (2.145) nazywamy *sterowalnym* w chwili  $t^\dagger$ . Układ sterowalny w dowolnie małej chwili czasu  $t > 0$  nazywamy sterowalnym w krótkim czasie. Okazuje się, że nieholonomiczność układu (2.145) jest warunkiem wystarczającym sterowalności w krótkim czasie przy pomocy odcinkami stałych sterowań. Oznacza to, że dla robota nieholonomicznego odwzorowanie  $\boldsymbol{\varphi}_t(\mathbf{q}_0, \cdot)$  jest suriekcją przy każdym  $\mathbf{q}_0$  i  $t > 0$ .

Niech zadanie realizowane przez robota mobilnego polega na osiągnięciu zadanego stanu  $\mathbf{q}_d$  w określonej chwili  $t \in [0, T]$  przy ustalonym stanie  $\mathbf{q}_0$  w chwili 0. Traktując stan  $\mathbf{q}_0$  jako ustalony, możemy zamiast (2.146) rozważyć odwzorowanie

$$\mathbf{k}_{\mathbf{q}_0, t} : \mathbb{L}_m^2[0, t] \longrightarrow \mathbb{R}^n, \quad \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot)) = \boldsymbol{\varphi}_t(\mathbf{q}_0, \mathbf{u}(\cdot)), \quad (2.147)$$

które nazwiemy kinematyką robota mobilnego w chwili  $t \in [0, T]$ . Dla odróżnienia, odwzorowanie  $\mathbf{k}_{\mathbf{q}_0, T}$  zdefiniowane przy pomocy formuły (2.147) dla  $t = T$  będziemy nazywać *kinematyką robota mobilnego*. Kinematyka  $\mathbf{k}_{\mathbf{q}_0, T}$  odwzorowuje  $\infty$ -wymiarową przestrzeń sterowań w przestrzeń stanu układu. Odpowiednikiem jakobianu analitycznego manipulatora jest w tym przypadku różniczka odwzorowania  $\mathbf{k}_{\mathbf{q}_0, T}$ . W celu jej obliczenia skorzystamy z kinematyki robota mobilnego w chwili  $t \in [0, T]$ . Różniczka<sup>‡</sup>  $D \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot))$ , w punkcie  $\mathbf{u}(\cdot) \in \mathbb{L}_m^2[0, t]$ , jest przekształceniem liniowym

$$D \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot)) : \mathbb{L}_m^2[0, t] \longrightarrow \mathbb{R}^n,$$

<sup>†</sup>Ściślej, osiągalnym, ale dla układów postaci (2.145) sterowalność jest równoważna osiągalności.

<sup>‡</sup>Różniczka Gâteaux.



określonym formułą

$$D \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot))\mathbf{v}(\cdot) = \left. \frac{d}{d\gamma} \right|_{\gamma=0} \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot) + \gamma\mathbf{v}(\cdot)). \quad (2.148)$$

Nietrudno zauważyć, że dla  $t = 0$  różniczka znika, tzn.  $D \mathbf{k}_{\mathbf{q}_0, 0}(\mathbf{u}(\cdot))\mathbf{v}(\cdot) = \mathbf{0}$ . Aby wyliczyć  $D \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot))$ , zróżniczkujemy obie strony (2.148) względem czasu

$$\begin{aligned} \frac{d}{dt} D \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot))\mathbf{v}(\cdot) &= \\ &= \left. \frac{d}{dt} \frac{d}{d\gamma} \right|_{\gamma=0} \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot) + \gamma\mathbf{v}(\cdot)) = \left. \frac{d}{d\gamma} \right|_{\gamma=0} \frac{d}{dt} \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot) + \gamma\mathbf{v}(\cdot)) = \\ &= \left. \frac{d}{d\gamma} \right|_{\gamma=0} \frac{d}{dt} \boldsymbol{\varphi}_t(\mathbf{q}_0, \mathbf{u}(\cdot) + \gamma\mathbf{v}(\cdot)) = \\ &= \left. \frac{d}{d\gamma} \right|_{\gamma=0} \mathbf{G}(\boldsymbol{\varphi}_t(\mathbf{q}_0, \mathbf{u}(\cdot) + \gamma\mathbf{v}(\cdot)))(\mathbf{u}(\cdot) + \gamma\mathbf{v}(\cdot))(t) = \\ &= \frac{\partial(\mathbf{G}(\mathbf{q}(t))\mathbf{u}(t))}{\partial \mathbf{q}} D \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot))\mathbf{v}(\cdot) + \mathbf{G}(\mathbf{q}(t))\mathbf{v}(t). \end{aligned}$$

Jak widać, różniczka  $D \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot))\mathbf{v}(\cdot)$  kinematyki  $\mathbf{k}_{\mathbf{q}_0, t}$  jest rozwiązaniem w chwili  $t$  tzw. równania wariacyjnego stowarzyszonego z układem (2.145). Równanie to przedstawia przybliżenie liniowe układu (2.145) wzdłuż pary (sterowanie, trajektoria) =  $(\mathbf{u}(t), \mathbf{q}(t))$  przy założeniu, że sterowanie uległo małej zmianie (wariacji) w kierunku  $\mathbf{v}(\cdot)$ . Przy oznaczeniach

$$\mathbf{A}(t) = \frac{\partial(\mathbf{G}(\mathbf{q}(t))\mathbf{u}(t))}{\partial \mathbf{q}}, \quad \mathbf{B}(t) = \mathbf{G}(\mathbf{q}(t)), \quad (2.149)$$

możemy je zapisać w następujący sposób

$$\begin{cases} \dot{\mathbf{q}}(t) = \mathbf{G}(\mathbf{q}(t))\mathbf{u}(t) \\ \frac{d}{dt} D \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot))\mathbf{v}(\cdot) = \mathbf{A}(t) D \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot))\mathbf{v}(\cdot) + \mathbf{B}(t)\mathbf{v}(t), \end{cases} \quad (2.150)$$

z warunkiem początkowym  $\mathbf{q}(0) = \mathbf{q}_0$  oraz  $D \mathbf{k}_{\mathbf{q}_0, t}(\mathbf{u}(\cdot))\mathbf{v}(\cdot)|_{t=0} = \mathbf{0}$ . Korzystając z układu równań (2.150) wyliczamy różniczkę  $D \mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))\mathbf{v}(\cdot)$  kinematyki w chwili  $T$

$$D \mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))\mathbf{v}(\cdot) = \int_0^T \boldsymbol{\Phi}(T, s)\mathbf{B}(s)\mathbf{v}(s) ds, \quad (2.151)$$

którą będziemy nazywać *jakobianem analitycznym* robota mobilnego. Macierz fundamentalna  $\Phi(t, s)$  występująca w formule (2.151) spełnia równanie

$$\frac{d}{dt}\Phi(t, s) = \mathbf{A}(t)\Phi(t, s)$$

przy warunku  $\Phi(s, s) = \mathbb{I}_n$ . Mówimy, że różniczka  $D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$  ma pełny rząd w punkcie  $\mathbf{u}(\cdot)$ , jeżeli  $D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$  jest suriekcją liniową<sup>§</sup>. Własność „ $D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$  ma pełny rząd” oznacza, że jeżeli za pośrednictwem sterowania  $\mathbf{u}(\cdot)$  osiągniemy stan  $\mathbf{q}$ , to każdy stan  $\mathbf{q}'$  z pewnego otoczenia  $\mathbf{q}$  możemy osiągnąć biorąc pewne sterowanie  $\mathbf{u}'(\cdot)$  bliskie  $\mathbf{u}(\cdot)$ . Sterowania  $\mathbf{u}(\cdot)$ , dla których rząd  $D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$  nie jest pełny, będziemy nazywać *sterowaniami osobliwymi*.

Sterowania osobliwe robota mobilnego będziemy nazywać *osobliwościami* kinematyki robota mobilnego. Analizę osobliwości kinematyki zaczniemy od bardzo prostego przykładu. Niech  $\mathbf{u}(\cdot) \equiv \mathbf{0}$ . Trajektoria układu (2.145) odpowiadająca temu sterowaniu jest, oczywiście, trajektorią stałą,  $\mathbf{q}(t) = \mathbf{q}_0$ . Z definicji (2.150) wynika, że macierz  $\mathbf{A}(t) \equiv \mathbf{0}$ , natomiast macierz  $\mathbf{B}(t) = \mathbf{G}(\mathbf{q}_0)$  jest stała. Oznacza to, że macierz fundamentalna  $\Phi(t, s) = \mathbb{I}_n$ , zaś jakobian analityczny

$$D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{0})\mathbf{v}(\cdot) = \mathbf{G}(\mathbf{q}_0) \int_0^T \mathbf{v}(s) ds = \mathbf{G}(\mathbf{q}_0)\mathbf{w}$$

dla pewnego  $\mathbf{w} \in \mathbb{R}^m$ . Łatwo zauważyć, że jeżeli  $m < n$ , to

$$\dim D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{0}) (\mathbb{L}_m^2[0, T]) \leq m < n,$$

a zatem sterowanie zerowe jest osobliwe w sposób trywialny.

W ogólnym przypadku warunki konieczne, jakie musi spełniać sterowanie osobliwe, otrzymujemy z warunków osobliwości sterowania w pewnym zadaniu sterowania optymalnego. Niech będzie dane zadanie sterowania optymalnego układu (2.145) z lagranżianem  $L(\mathbf{q}, \mathbf{u})$ . Zgodnie z wymaganiami Zasady Maksimum Pontriagina utwórzmy hamiltonian

$$H(\mathbf{q}, \mathbf{p}, p_0, \mathbf{u}) = \mathbf{p}^T \mathbf{G}(\mathbf{q})\mathbf{u} - p_0 L(\mathbf{q}, \mathbf{u}), \quad (2.152)$$

w którym współczynnik  $p_0$  może być równy zero. Sterowanie optymalne  $\mathbf{u}(t)$  nazywamy osobliwym, jeżeli  $p_0$  oraz odpowiadająca mu ekstremala

<sup>§</sup>Tzn. wymiar obrazu  $\dim D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))(\mathbb{L}_m^2[0, T]) = n$ .

jest *osobliwa*<sup>†</sup>. W tym celu, dla  $i = 1, 2, \dots, m$ , muszą być spełnione warunki

$$\mathbf{p}^T \mathbf{g}_i(\mathbf{q}) = 0. \quad (2.153)$$

Przypominamy, że zmienna dołączona nie znika, tzn.  $\mathbf{p} \neq \mathbf{0}$ . Aby uzyskać warunki konieczne, jakie musi spełniać ekstremala osobliwa (a więc, a fortiori, także sterowanie osobliwe), różniczkujemy warunek (2.153) względem czasu

$$\frac{d}{dt} \mathbf{p}^T \mathbf{g}_i(\mathbf{q}) = \mathbf{p}^T \sum_{j=1}^m [\mathbf{g}_j, \mathbf{g}_i](\mathbf{q}) \mathbf{u}_j = \mathbf{p}^T [\mathbf{X}_u, \mathbf{g}_i](\mathbf{q}) = 0, \quad (2.154)$$

gdzie symbolem  $[\mathbf{X}, \mathbf{Y}]$  oznaczyliśmy nawias Liego pól wektorowych, natomiast pole wektorowe  $\mathbf{X}_u(\mathbf{q}) = \mathbf{G}(\mathbf{q})\mathbf{u}$ . Biorąc równania (2.154) dla  $i = 1, 2, \dots, m$ , otrzymujemy zależność

$$\mathbf{M}(\mathbf{q}, \mathbf{p})\mathbf{u} = \mathbf{0}, \quad (2.155)$$

w której macierz  $\mathbf{M}(\mathbf{q}, \mathbf{p})$  o elementach  $m_{ij}(\mathbf{q}, \mathbf{p}) = \mathbf{p}^T [\mathbf{g}_j, \mathbf{g}_i](\mathbf{q})$  jest skośnie symetryczna. Jeżeli ta macierz jest nieosobliwa, jedynym sterowaniem osobliwym jest sterowanie zerowe  $\mathbf{u}(\cdot) \equiv \mathbf{0}$ . W przeciwnym wypadku, poprzez wytrwałe różniczkowanie zależności (2.154) wyprowadzamy kolejny warunek konieczny

$$\mathbf{M}(\mathbf{q}, \mathbf{p})\dot{\mathbf{u}} = \mathbf{N}(\mathbf{q}, \mathbf{p}, \mathbf{u}),$$

w którym  $i$ -ta składowa wektora  $\mathbf{N}(\mathbf{q}, \mathbf{p}, \mathbf{u})$  wyraża się wzorem

$$N_i(\mathbf{q}, \mathbf{p}, \mathbf{u}) = \sum_{j,k} \mathbf{p}^T [\mathbf{g}_k, [\mathbf{g}_j, \mathbf{g}_i]](\mathbf{q}) \mathbf{u}_j \mathbf{u}_k.$$

Kontynuując różniczkowanie, można wyprowadzić warunki konieczne wyższych rzędów na sterowanie osobliwe. Do tematu sterowań osobliwych wrócimy jeszcze raz w podrozdziale 4.1.

Na zakończenie wywodów formalnych przedstawimy prosty przykład wyznaczenia sterowań osobliwych przy pomocy wyprowadzonych powyżej warunków koniecznych.

#### Przykład 2.4.1 (Sterowanie osobliwe)

Niech będzie dany układ sterowania

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u} = \begin{pmatrix} -q_2 u_1 \\ u_2 \\ u_1 \\ u_3 \end{pmatrix} = \begin{pmatrix} -q_2 \\ 0 \\ 1 \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} u_2 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_3, \quad (2.156)$$

<sup>†</sup>Zobacz dodatek A.5.

gdzie  $\mathbf{q} \in \mathbb{R}^4$ ,  $\mathbf{u} \in \mathbb{R}^3$ . Z warunku (2.153) wynika, że

$$p_3 - p_1 q_2 = 0 \quad \text{oraz} \quad p_2 = p_4 = 0. \quad (2.157)$$

Nietrudno wyliczyć macierz

$$\mathbf{M}(\mathbf{q}, \mathbf{p}) = \begin{bmatrix} 0 & -p_1 & 0 \\ p_1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = p_1 [\mathbf{e}_3],$$

co pozwala napisać równania (2.155)

$$p_1 u_2 = 0 \quad \text{i} \quad p_1 u_1 = 0. \quad (2.158)$$

Zauważmy, że ze względu na nieznikanie zmiennej dołączonej, z zależności (2.157) wynika  $p_1 \neq 0$ , co w połączeniu z (2.158) daje  $u_1 = u_2 = 0$ . W konsekwencji, warunki konieczne są spełnione przez sterowanie osobliwe  $\mathbf{u}(t) = (0, 0, u_3(t))^T$  przy dowolnym  $u_3(t)$ . ■

## 2.5 Komentarze i uwagi bibliograficzne

Podstawowe pojęcia kinematyki punktu materialnego i ciała sztywnego można znaleźć we wstępnych rozdziałach podręczników do mechaniki analitycznej [Gut71, Wit77, RK95] lub robotyki [Pau81, AS86, Cra93, Lat93, MLS94, Duf96, SS96, SV97]. Konsekwentne, geometryczne ujęcie kinematyki ciała sztywnego w języku teorii grup i algebr Liego zawiera monografia [MLS94]. Podstawowe pojęcia geometryczne stosowane w robotyce zostały wyłożone w sposób wyczerpujący w pracy [Sel96]. Parametryzacje i układy współrzędnych w specjalnej grupie obrotów oraz w specjalnej grupie euklidesowej należą do klasycznego materiału podręcznikowego [GR84, Cra93, Lat93, MLS94, SV97, Ang97]. Oceny różnych parametryzacji z punktu widzenia dokładności śledzenia trajektorii dokonano ostatnio w pracy [C<sup>+</sup>98]. Formułę definiującą współrzędne wykładnicze w  $\text{SE}(3)$  zaczerpnęliśmy z pracy [TD94]. Termin „więzy nieholonomiczne” został wprowadzony przez Hertza, [AKN88]. Postać ograniczeń fazowych zwana postacią Pfaffa wywodzi się z teorii równań różniczkowych cząstkowych [Zhi92]. Dynamiki układów nieholonomicznych dotyczą prace [NF71] oraz [VG94]. Pojęcia różniczkowej dystrybucji oraz nawiasu Liego należą do geometrii różniczkowej [Gan87, Spi79]. Narzędzia geometrii różniczkowej w zakresie używanym do opisu i analizy kinematyki układów

nieholonomicznych zostały rozwinięte w obrębie geometrycznej teorii sterowania [Isi89, NS90]. Warunek wystarczający nieholonomiczności więzów przedstawiony w twierdzeniu 2.2.1 oznacza, że ruch układu nie może być ograniczony do żadnej właściwej podrozmaitości uniwersum konfiguracyjnego i jest równoważny warunkowi sterowalności układu (2.33), znanemu jako twierdzenie Chow [Cho39, Isi89, NS90]. Warunek holonomiczności ograniczeń fazowych wynika z twierdzenia Frobeniusa [Spi79] i orzeka, że ruch układu jest ograniczony do pewnej podrozmaitości uniwersum konfiguracyjnego. W przypadku częściowej holonomiczności kowymiar tej podrozmaitości może być mniejszy niż liczba ograniczeń fazowych. Więzy nazywane przez nas nieholonomicznymi bywają w literaturze określane mianem całkowicie nieholonomicznych [MLS94]. Punkty, w których otoczeniu stopień nieholonomiczności dystrybucji ulega zmianie, nazywają się osobliwościami dystrybucji. Klasyfikację osobliwości dystrybucji robota mobilnego złożonego z ciągnika i przyczep podaje praca [Jea96]. Ze względu na to, że kinematyka manipulatora jest odwzorowaniem rozmaitości, staramy się stosować terminy „rozmaitość przegubowa” i „rozmaitość zadaniowa”, w odróżnieniu od przestrzeni przegubowej i przestrzeni zadaniowej, które pojawiają się po zdefiniowaniu na tych rozmaitościach układów współrzędnych lub parametryzacji. Reprezentacja Denavita-Hartenberga należy do znanych, standardowych narzędzi analizy kinematyki manipulatorów [DH55, Pau81, SV97, Ang97, CD98]. Reprezentacja używana przez nas nosi nazwę standardowej; alternatywą jest tzw. reprezentacja zmodyfikowana. Idea reprezentacji wykładniczej kinematyki manipulatora pochodzi od Brocketta [Bro84]; została ona następnie rozwinięta w książce [MLS94]. Przykładem reprezentacji algebraicznej kinematyki manipulatora jest reprezentacja kwaternionowa (lub bikwaternionowa) wykorzystująca własność nakrycia grupy  $\mathbb{S}\mathbb{O}(3)$  przez kwaterniony jednostkowe [Sel96, Sic98]. Pojęcie skrętnika należy do teorii śrub, której podstawy można znaleźć w pracach [WN92, MLS94]. Termin „jakobian geometryczny” pochodzi od Siciliano [SS96] i oznacza transformację prędkości manipulatora, której wyliczenie nie wymaga wykorzystania układów współrzędnych przegubowych ani zadaniowych. Przez odwołanie się do formuły prędkości efektora w układzie przestrzeni i w układzie ciała zdefiniowaliśmy jakobian geometryczny w przestrzeni i jakobian geometryczny w ciele. Obiekt geometryczny zwany w literaturze jakobianem manipulatora, [Pau81, SV97], nie jest żadnym z wymienionych jakobianów geometrycznych. Związki między jakobianami analitycznymi a geometrycznymi są konsekwencją formuły transformacji

prędkości (2.119). Formuła ta może również posłużyć do analizy osobliwości reprezentacji manipulatora. O osobliwościach reprezentacji traktuje praca [Dul96]. Zastosowanie reprezentacji wykładniczej kinematyki do obliczania jacobianu geometrycznego wydaje się przekonujące. Do najciekawszych aspektów analizy kinematyki manipulatorów należy zadanie wyznaczenia konfiguracji osobliwych [Cra93, MLS94, SV97]. Badania konfiguracji osobliwych manipulatorów zainicjował Whitney [Whi72]. Problematyka osobliwości kinematycznych jest uprawiana przez autorów tej książki od wielu lat [Tch90, Tch91, TU92, TD93, Tch95, Mus96, MT96, TM97, TM98, Tch98]. Opis kinematyki robota mobilnego jako odwzorowania osiągalności stanów pewnego układu sterowania pochodzi z prac Wena i współpracowników [DW94, PW96, DW97a, DW97b, DSW98], i pozwala uzyskać reprezentację kinematyki robota mobilnego formalnie analogiczną do reprezentacji kinematyki manipulatora. Wymieniona analogia rozciąga się na pojęcia jacobianu analitycznego oraz osobliwości kinematyki. Przy wyprowadzeniu formuły jacobianu analitycznego (2.151) można skorzystać z wyników przedstawionych w książce Sontaga [Son90]. Zadanie sterowania optymalnego (2.152) było rozważane w pracy [Mon92] i monografii [MLS94]; tam też wyprowadzono warunek (2.155). Więcej informacji na temat sterowań osobliwych w kontekście Zasady Maksimum Pontriagina można znaleźć w książce [VG97]. Przykładowy układ sterowania (2.156) został zdefiniowany za pośrednictwem tzw. struktury quasi-kontaktowej w  $\mathbb{R}^4$ .

## Literatura

- [AKN88] V. I. Arnold, V. V. Kozlov i A. I. Neishtadt, *Mathematical aspects of classical and celestial mechanics*. W: V. I. Arnold, (red.), *Dynamical Systems III*. Springer-Verlag, Berlin, 1988.
- [Ang97] J. Angeles, *Fundamentals of Robotic Mechanical Systems. Theory, Methods, and Algorithms*. Springer-Verlag, New York, 1997.
- [AS86] H. Asada i J. J. E. Slotine, *Robot Analysis and Control*. J. Wiley and Sons, New York, 1986.
- [Bro84] R. W. Brockett, *Robotic manipulators and the product of exponentials formula*. W: *Mathematical Theory of Networks and Systems*, strony 120–129. Springer-Verlag, Berlin, 1984.
- [C<sup>+</sup>98] F. Caccavale et al., *Resolved-acceleration control of robot manipulators: A critical review with experiments*. *Robotica*, **16**:565–573, 1998.

- [CD98] C. D. Crane III i J. Duffy, *Kinematic Analysis of Robot Manipulators*. Cambridge University Press, Cambridge, 1998.
- [Cho39] W. L. Chow, Über Systeme von linearen partiellen Differentialgleichungen erster Ordnung. *Math. Ann.*, **117**(1):98–105, 1939.
- [Cra93] J. J. Craig, *Wprowadzenie do robotyki*. WNT, Warszawa, 1993.
- [DH55] J. Denavit i R. S. Hartenberg, A kinematic notation for lower-pair mechanisms based on matrices. *ASME Trans. J. Applied Mechanics*, **77**(2):215–221, 1955.
- [DSW98] A. W. Divilbiss, S. Seereeram i J. T. Wen, Kinematic path planning for robots with holonomic and nonholonomic constraints. W: J. Bailleul, S. S. Sastry i H. J. Sussmann, (red.), *Essays on Mathematical Robotics*, strony 127–150. Springer-Verlag, New York, 1998.
- [Duf96] J. Duffy, *Statics and Kinematics with Application to Robotics*. Cambridge University Press, Cambridge, 1996.
- [Dul96] I. Dułęba, O unkianiu osobliwości reprezentacji. W: *Symp. Symul. Proc. Dynam.*, strony 345–350, Polana Chochołowska, 1996.
- [DW94] A. W. Divilbiss i J. T. Wen, Nonholonomic path planning with inequality constraints. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 52–57, San Diego, 1994.
- [DW97a] A. W. Divilbiss i J. T. Wen, A path space approach to nonholonomic motion planning in the presence of obstacles. *IEEE Trans. Robotics Automat.*, **13**(3):443–451, 1997.
- [DW97b] A. W. Divilbiss i J. T. Wen, Trajectory tracking control for a car-trailer system. *IEEE Trans. Control Syst. Technology*, **5**(3):269–278, 1997.
- [Gan87] J. Gancarzewicz, *Geometria różniczkowa*. PWN, Warszawa, 1987.
- [GR84] B. Gorla i M. Renaud, *Modèles des robots manipulateurs*. Capadues Publishers, Toulouse, 1984.
- [Gut71] R. Gutowski, *Mechanika analityczna*. PWN, Warszawa, 1971.
- [Isi89] A. Isidori, *Nonlinear Control Systems*. Springer-Verlag, Berlin, 1989.
- [Jea96] F. Jean, The car with n-trailers: Characterisation of the singular configurations. *Control, Optimisation and Calculus of Variations*, **1**:241–266, 1996.
- [Lat93] J. C. Latombe, *Robot Motion Planning*. Kluwer, Boston, 1993.
- [MLS94] R. M. Murray, Z. Li i S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton, 1994.

- [Mon92] R. Montgomery, Abnormal optimal controls and open problems in non-holonomic steering. W: *Proc. NOLCOS Symposium*, vol. 1, strony 373–378, Bordeaux, 1992.
- [MT96] R. Muszyński i K. Tchoń, Normal forms of non-redundant singular robot kinematics: Three DOF worked examples. *J. Robotic Systems*, **13**(12):765–791, 1996.
- [Mus96] R. Muszyński, *Modele i algorytmy sterowania manipulatorów z osobliwościami kinematycznymi*. Rozprawa doktorska, Instytut Cybernetyki Technicznej, Politechnika Wrocławska, 1996.
- [NF71] J. Nejmank i N. Fufajew, *Dynamika układów nieholonomicznych*. PWN, Warszawa, 1971.
- [NS90] H. Nijmeijer i A. J. van der Schaft, *Nonlinear Dynamic Control Systems*. Springer-Verlag, New York, 1990.
- [Pau81] R. P. Paul, *Robot Manipulators: Mathematics, Programming and Control*. MIT Press, Cambridge, 1981.
- [PW96] D. O. Popa i J. T. Wen, Nonholonomic path-planning with obstacle avoidance: A path-space approach. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 4, strony 2662–2667, Minneapolis, 1996.
- [RK95] W. Rubinowicz i W. Królikowski, *Mechanika teoretyczna*. PWN, Warszawa, 1995.
- [Sel96] J. M. Selig, *Geometrical Methods in Robotics*. Springer-Verlag, New York, 1996.
- [Sic98] B. Siciliano, On the use of quaternions for robot interaction control tasks. W: *Proc. MMAR Symposium*, vol. 3, strony 739–746, Międzyzdroje, 1998.
- [Son90] E. D. Sontag, *Mathematical Control Theory*. Springer-Verlag, New York, 1990.
- [Spi79] M. Spivak, *A Comprehensive Introduction to Differential Geometry*. Publish or Perish, Berkeley, 1979.
- [SS96] L. Sciavicco i B. Siciliano, *Modeling and Control of Robot Manipulators*. The McGraw-Hill, New York, 1996.
- [SV97] M. Spong i M. Vidyasagar, *Dynamika i sterowanie robotów*. WNT, Warszawa, 1997.
- [Tch90] K. Tchoń, Towards a differential topological classification of robot manipulators. W: *Robust Control of Linear Systems and Nonlinear Control*, strony 565–574. Birkhäuser, Boston, 1990.



- [Tch91] K. Tchoń, Differential topology of the inverse kinematic problem for redundant robot manipulators. *Int. J. Robotics Research*, **10**(5):492–504, 1991.
- [Tch95] K. Tchoń, A normal form of singular kinematics of robot manipulators with smallest degeneracy. *IEEE Trans. Robotics Automat.*, **11**(3):401–404, 1995.
- [Tch98] K. Tchoń, Quadratic normal forms of redundant robot kinematics with application to singularity avoidance. *IEEE Trans. Robotics Automat.*, **14**(5):834–837, 1998.
- [TD93] K. Tchoń i I. Dulęba, On inverting singular kinematics and geodesic trajectory generation for robot manipulators. *J. Intelligent Robotic Systems*, **8**:325–359, 1993.
- [TD94] K. Tchoń i I. Dulęba, Definition of a kinematic metric for robot manipulators. *J. Robotic Systems*, **11**(3):211–222, 1994.
- [TM97] K. Tchoń i R. Muszyński, Singularities of non-redundant robot kinematics. *Int. J. Robotics Research*, **16**(1):60–76, 1997.
- [TM98] K. Tchoń i R. Muszyński, Singular inverse kinematic problem for robotic manipulators: A normal form approach. *IEEE Trans. Robotics Automat.*, **14**(1):93–104, 1998.
- [TU92] K. Tchoń i P. Urban, Classification of kinematic singularities in planar robot manipulators. *Systems & Contr. Lett.*, **19**(3):293–302, 1992.
- [VG94] A. M. Vershik i V. Ya. Gershkovich, Nonholonomic dynamical systems, geometry of distribution and variational problems. W: V. I. Arnold, (red.), *Dynamical Systems VII*, strony 1–81. Springer-Verlag, Berlin, 1994.
- [VG97] T. L. Vincent i W. J. Grantham, *Nonlinear and Optimal Control Systems*. J. Wiley and Sons, New York, 1997.
- [Whi72] D. E. Whitney, The mathematics of coordinated control of prosthetic arms and manipulators. *J. Dyn. Syst. Meas. Contr.*, **94** G(4):303–309, 1972.
- [Wit77] J. Wittenburg, *Dynamics of Systems of Rigid Bodies*. Teuber, Stuttgart, 1977.
- [WN92] J. Wojnarowski i A. Nowak, *Mechanika manipulatorów-robotów w opisie motorów*. Wydawnictwo Politechniki Śląskiej, Gliwice, 1992.
- [Zhi92] M. Zhitomirski, *Typical Singularities of Differential 1-Forms and Pfaffian Equations*. AMS, Providence, 1992.



## Rozdział 3

# Algorytmy kinematyki odwrotnej manipulatora

Niech będzie dany manipulator o  $n$  stopniach swobody, z rozmainością przegubową  $\mathcal{Q}$  i rozmainością zadaniową  $\mathcal{Z} \subset \text{SE}(3)$ . Zadanie polegające na wyznaczeniu kinematyki

$$\mathbf{K} : \mathcal{Q} \longrightarrow \mathcal{Z} \subset \text{SE}(3).$$

manipulatora, w postaci reprezentacji Denavita-Hartenberga lub reprezentacji wykładniczej, nazywa się *prostym (bezpośrednim) zadaniem kinematyki*. W wyniku jego rozwiązania otrzymujemy zależność położenia i orientacji efektora manipulatora od konfiguracji jego przegubów. Prostemu zadaniu kinematyki poświęciliśmy wiele miejsca w poprzednim rozdziale. Z punktu widzenia sterowania manipulatora ważniejsze jest jednak inne zadanie, polegające na wyznaczeniu ruchu przegubów manipulatora zapewniającego wykonanie określonego ruchu efektora na rozmainości zadaniowej. To ostatnie zadanie nazywa się *odwrotnym zadaniem kinematyki* i formułuje w następujący sposób:

Mając daną trajektorię ruchu  $\mathbf{z}_d : \mathcal{I} \longrightarrow \mathcal{Z}$  na rozmainości zadaniowej, określoną na pewnym przedziale czasu  $\mathcal{I} \subseteq \mathbb{R}$ , wyznaczyć odpowiadającą jej trajektorię ruchu przegubów  $\mathbf{q}_d : \mathcal{I} \longrightarrow \mathcal{Q}$ , taką że dla  $t \in \mathcal{I}$

$$\mathbf{K}(\mathbf{q}_d(t)) = \mathbf{z}_d(t). \quad (3.1)$$

Sformułowane przez nas odwrotne zadanie kinematyki nazywa się *zadaniem ciągłym*. W szczególnym przypadku, zadanie manipulatora może polegać

na osiągnięciu jednego, określonego położenia i orientacji efektora  $\mathbf{z}_d \in \mathcal{Z}$ . Takie zadanie odwrotne nazywamy *punktowym*. Rozwiązaniem zadania punktowego jest położenie przegubów  $\mathbf{q}_d$ , takie że  $\mathbf{K}(\mathbf{q}_d) = \mathbf{z}_d$ .

Zadanie odwrotne może być w naturalny sposób sformułowane we współrzędnych zadaniowych i przegubowych. Przy sformułowaniu we współrzędnych, mając daną trajektorię  $\mathbf{y}_d(t)$  w przestrzeni zadaniowej i kinematykę  $\mathbf{k} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ , należy wyznaczyć trajektorię  $\mathbf{x}_d(t)$  w przestrzeni przegubowej, taką że dla  $t \in \mathcal{I}$

$$\mathbf{y}_d(t) = \mathbf{k}(\mathbf{x}_d(t)). \quad (3.2)$$

Z matematycznego punktu widzenia, odwrotne zadanie kinematyki polega na rozwiązaniu układu równań nieliniowych (3.2) ze względu na wektor współrzędnych przegubowych  $\mathbf{x}_d(t)$  w każdej chwili  $t \in \mathcal{I}$ . Oczywistym warunkiem koniecznym istnienia rozwiązania jest żądanie, by trajektoria przeznaczona do realizacji w przestrzeni zadaniowej była zawarta w zbiorze wartości odwzorowania  $\mathbf{k}$ . Zbiór ten będziemy nazywać *przestrzenią roboczą* manipulatora,  $\mathcal{W} = \mathbf{k}(\mathbb{R}^n)^*$ . Jeżeli występują ograniczenia ruchu przegubów, przestrzeń robocza jest obrazem dopuszczalnych położań przegubów. Zauważmy, że w przypadku gdy wymiar  $m$  przestrzeni zadaniowej jest większy od liczby stopni swobody manipulatora, przestrzeń robocza będzie stanowić „mały” podzbiór przestrzeni zadaniowej, a zatem tylko bardzo szczególne trajektorie zadaniowe znajdują się w przestrzeni roboczej manipulatora i uzyskują atrybut realizowalności. Załóżmy, że zadana trajektoria  $\mathbf{y}_d(t) \in \mathcal{W}$  jest gładką funkcją czasu. Wówczas odwrotne zadanie kinematyki posiada rozwiązanie dla każdej chwili  $t \in \mathcal{I}$ , ale rozwiązanie to ani nie musi być jednoznaczne, ani nie musi zależeć w sposób gładki od czasu. Jeżeli manipulator jest nieredundantny, liczba rozwiązań zadania odwrotnego może być skończona lub nieskończona, zależnie od typu konfiguracji osobliwych manipulatora. Dla manipulatora redundantnego zbiór rozwiązań zadania odwrotnego jest z reguły mocy continuum.

Zgodnie z klasyfikacją przeprowadzoną w podrozdziale 2.3.5, wszystkie konfiguracje, jakie może przyjąć manipulator w trakcie ruchu, dzielą się na regularne i osobliwe. Analogiczny podział można wprowadzić w obrębie odwrotnych zadań kinematyki. Powiemy, że odwrotne zadanie kinematyki

---

\*Nasza definicja przestrzeni roboczej jest szersza od definicji podręcznikowej, w myśl której przestrzeń robocza składa się wyłącznie z osiągalnych położań (nie orientacji) efektora.

jest regularne, jeśli realizacja trajektorii ruchu efektora nie wymaga wprowadzenia manipulatora w konfiguracje osobliwe. W przeciwnym wypadku odwrotne zadanie kinematyki będziemy nazywali osobliwym.

### 3.1 Regularne odwrotne zadanie kinematyki

Przedstawimy teraz metody rozwiązania regularnego odwrotnego zadania kinematyki. Możliwe strategie rozwiązania tego zadania są dwojakie: symboliczne (dające rozwiązanie w postaci jawnej) i numeryczne. Rozwiązanie zadania w postaci jawnej uzyskujemy dzięki analizie struktury manipulatora i opisujących go równań kinematyki, prowadzącej do wyrażenia kinematyki odwrotnej manipulatora w postaci symbolicznej

$$\mathbf{q} \in \mathbf{K}^{-1}(\mathbf{z}) \quad (3.3)$$

dla kinematyki opisanej przez (2.40), lub jako

$$\mathbf{x} \in \mathbf{k}^{-1}(\mathbf{y}) \quad (3.4)$$

dla kinematyki postaci (2.65)\*. Wyznaczenie jednego z powyższych wyrażeń wzdłuż zadanej trajektorii ruchu w przestrzeni zadaniowej pozwala na znalezienie odpowiadającej jej trajektorii ruchu przegubów. Rozwiązanie odwrotnego zadania kinematyki w postaci numerycznej wymaga wprowadzenia układów współrzędnych zadaniowych i przegubowych, i polega na przedstawieniu zależności między prędkościami ruchu w przestrzeni zadaniowej i przegubowej manipulatora w postaci układu równań różniczkowych. Przykładowo, dla manipulatora nieredundantnego będzie to układ równań

$$\dot{\mathbf{x}} = (\mathbf{J}^a)^{-1}(\mathbf{x})\dot{\mathbf{y}}, \quad (3.5)$$

gdzie  $\mathbf{J}^a(\mathbf{x})$  oznacza jacobian analityczny manipulatora. Wynikową trajektorię przegubową wyliczamy przez scałkowanie układu (3.5) dla konkretnej trajektorii zadanej zawartej w przestrzeni roboczej. W podobny sposób można uzyskać rozwiązanie odwrotnego zadania kinematyki z wykorzystaniem innych niż analityczny jacobianów manipulatora.

Rozwiązania w postaci symbolicznej, aczkolwiek zazwyczaj trudne do znalezienia, mają tę przewagę nad rozwiązaniami numerycznymi, że nie wymagają wielokrotnego powtarzania obliczeń. Do ich znalezienia wystarcza

---

\*Zapisy (3.3), (3.4) oznaczają, że otrzymany układ równań kinematyki odwrotnej może mieć wiele rozwiązań i należy dokonać wyboru jednego z nich.

bowiem jednorazowe określenie odpowiednich wyrażeń symbolicznych wyliczanych następnie wzdłuż różnych trajektorii zadanych. W przeciwieństwie do tego, rozwiązanie w postaci numerycznej wymaga całkowania układu równań różniczkowych (3.5) dla każdej z zadanych trajektorii, przy użyciu numerycznych metod iteracyjnych.

W dalszym ciągu, w dwóch kolejnych podrozdziałach, przedstawimy metody rozwiązywania regularnego odwrotnego zadania kinematyki manipulatora, zaliczane do klasy metod symbolicznych. Trzeci podrozdział poświęćmy opisowi kilku metod jakobianowych pozwalających na numeryczne rozwiązanie zadania odwrotnego.

### 3.1.1 Bezpośrednie podejście algebraiczne

Rozwiązanie odwrotnego zadania kinematyki przy zastosowaniu bezpośredniego podejścia algebraicznego polega na formalnym przekształceniu równań kinematyki manipulatora do postaci zależności opisujących odwzorowanie odwrotne do kinematyki. Ze względu na nieliniowość równań oraz zazwyczaj nieuniknioną niejednoznaczność rozwiązań, realizacja takiego podejścia często napotyka na trudności. Zastosowanie bezpośredniego podejścia algebraicznego zilustrujemy na przykładach.

#### Przykład 3.1.1 (Manipulator typu podwójne wahadło)

Rozważmy kinematykę (2.68) manipulatora typu podwójne wahadło. Podniesienie do kwadratu składowych wektora  $\mathbf{y}$  i ich dodanie prowadzi do równania

$$y_1^2 + y_2^2 = l_1^2 + l_2^2 + 2l_1l_2c_2,$$

z którego otrzymujemy

$$c_2 = \frac{y_1^2 + y_2^2 - l_1^2 - l_2^2}{2l_1l_2}. \quad (3.6)$$

Widzimy, że rozwiązanie zadania istnieje tylko wtedy, gdy prawa strona równości (3.6) przyjmuje wartości w przedziale  $[-1, 1]$ ; w przeciwnym przypadku punkt docelowy leżałby poza przestrzenią roboczą manipulatora. Gdy punkt ten znajduje się wewnątrz przestrzeni roboczej, możemy skorzystać z tożsamości

$$s_2 = \pm \sqrt{1 - c_2^2} \quad (3.7)$$

i wyliczyć kąt  $\alpha_2^*$

$$\alpha_2 = \text{atan2}(s_2, c_2). \quad (3.8)$$

W zależności od wyboru znaku we wzorze (3.7) otrzymujemy dwa rozwiązania. Przedstawiony schemat postępowania (znalezienie sinusa i cosinusa kąta, a następnie wyliczenie kąta przy użyciu funkcji  $\text{atan2}$ ) jest często wykorzystywany przy wyznaczaniu symbolicznej postaci odwzorowania opisującego kinematykę odwrotną.

Zajmiemy się teraz wyliczeniem składowej  $\alpha_1$  wektora  $\mathbf{x}$ . Wykorzystując wzory na sinus i cosinus sumy kątów i oznaczając  $\bar{l}_1 = l_1 + l_2 c_2$ ,  $\bar{l}_2 = l_2 s_2$ , przekształcamy równania kinematyki (2.68) do postaci

$$\mathbf{y} = \begin{pmatrix} \bar{l}_1 c_1 - \bar{l}_2 s_1 \\ \bar{l}_1 s_1 + \bar{l}_2 c_1 \end{pmatrix},$$

przy oznaczeniach  $r = \sqrt{\bar{l}_1^2 + \bar{l}_2^2}$  i  $\gamma = \text{atan2}(\bar{l}_2, \bar{l}_1)$ ,  $\bar{l}_1 = r \cos \gamma$ ,  $\bar{l}_2 = r \sin \gamma$ . Korzystając z tego otrzymujemy

$$\mathbf{y} = \begin{pmatrix} r \cos \gamma c_1 - r \sin \gamma s_1 \\ r \cos \gamma s_1 + r \sin \gamma c_1 \end{pmatrix} = \begin{pmatrix} r \cos(\gamma + \alpha_1) \\ r \sin(\gamma + \alpha_1) \end{pmatrix},$$

co w dalszym ciągu pozwala na wyliczenie kąta  $\gamma + \alpha_1$  w postaci

$$\gamma + \alpha_1 = \text{atan2}\left(\frac{y_2}{r}, \frac{y_1}{r}\right) = \text{atan2}(y_2, y_1),$$

a stąd

$$\alpha_1 = \text{atan2}(y_2, y_1) - \text{atan2}(\bar{l}_2, \bar{l}_1). \quad (3.9)$$

Zauważmy, że wybór znaku w (3.7) wpływa za pośrednictwem  $\bar{l}_1$  i  $\bar{l}_2$  na wyliczoną wartość  $\alpha_1$ . Zestawiając razem wyrażenia (3.9) i (3.8) otrzymamy symboliczne równania kinematyki odwrotnej manipulatora typu podwójne wahadło w aspekcie współrzędnych  $X_0 Y_0$ , w postaci

$$\mathbf{x} = \begin{pmatrix} \text{atan2}(y_2, y_1) - \text{atan2}(l_2 s_2, l_1 + l_2 c_2) \\ \text{atan2}(s_2, c_2) \end{pmatrix}, \quad (3.10)$$

gdzie  $c_2 = \frac{y_1^2 + y_2^2 - l_1^2 - l_2^2}{2l_1 l_2}$ , a  $s_2 = \pm \sqrt{1 - c_2^2}$ .

---

\*Funkcja  $\text{atan2}(x, y)$  wylicza  $\text{arctg } x/y$  z uwzględnieniem znaków przy  $x$  i  $y$  w celu wyznaczenia ćwiartki, w której leży wyliczany kąt: przy  $x, y$  dodatnich jest to pierwsza ćwiartka, przy  $x$  ujemnym,  $y$  dodatnim — druga ćwiartka, przy  $x, y$  ujemnych — trzecia, a przy  $x$  dodatnim,  $y$  ujemnym — czwarta.

Dla kinematyki manipulatora typu podwójne wahadło określonej w aspekcie współrzędnych  $\times_0\varphi$  równaniami (2.69), znalezienie wyrażeń symbolicznych opisujących kinematykę odwrotną jest znacznie prostsze. Po podstawieniu drugiego równania kinematyki do równania pierwszego dostajemy

$$y_1 = l_1 c_1 + l_2 \cos y_2,$$

skąd obliczamy  $c_1 = \frac{y_1 - l_2 \cos y_2}{l_1}$ . Następnie, biorąc  $s_1 = \pm \sqrt{1 - c_1^2}$  otrzymujemy

$$x_1 = \text{atan2}(s_1, c_1).$$

Wykorzystanie drugiej składowej kinematyki (2.69) pozwala zapisać równania kinematyki odwrotnej manipulatora w aspekcie współrzędnych  $\times_0\varphi$  jako

$$\mathbf{x} = \begin{pmatrix} \text{atan2}(s_1, c_1) \\ y_2 - \text{atan2}(s_1, c_1) \end{pmatrix} \quad (3.11)$$

z funkcjami  $s_1$  i  $c_1$  zdefiniowanymi jak wyżej. ■

### Przykład 3.1.2 (Manipulator IRb-6 na torze jezdny)

Rozpatrzmy kinematykę manipulatora IRb-6 opisaną równaniami (2.76). Rozwiązanie trzech ostatnich równań ze względu na składowe wektora  $\mathbf{x}$  jest natychmiastowe

$$\begin{cases} x_2 = -y_4 \\ x_5 = \pi - y_5 \\ x_6 = \pi - y_6. \end{cases} \quad (3.12)$$

Pozostałe trzy równania kinematyki (2.76) mogą zostać przedstawione w postaci

$$\begin{cases} a_2 c_3 + a_3 c_4 = \frac{y_1}{\cos y_4} - d_6 \sin y_5 \\ a_2 s_3 + a_3 s_4 = -d_6 \cos y_5 - d_1 - y_2 \\ x_1 = y_3 + \sin y_4 (a_2 c_3 + a_3 c_4 + d_6 \sin y_5). \end{cases} \quad (3.13)$$

Przy oznaczeniu wyrażeń niezależnych od zmiennych przegubowych jako  $w_1 = \frac{y_1}{\cos y_4} - d_6 \sin y_5$  oraz  $w_2 = -d_6 \cos y_5 - d_1 - y_2$ , łatwo zauważyć, że pierwsze dwa równania układu (3.13) opisują kinematykę manipulatora typu



podwójne wahadło w aspekcie współrzędnych położenia, ze współrzędnymi przegubowymi  $(x_3, x_4 - x_3)$  i zadaniowymi  $(w_1, w_2)$ <sup>†</sup>. Dla takiej kinematyki równania kinematyki odwrotnej są dane przez<sup>‡</sup>

$$\begin{cases} x_3 = \operatorname{atan2}(w_2, w_1) - \operatorname{atan2}(a_3 s_{4-3}, a_2 + a_3 c_{4-3}) \\ x_4 = \operatorname{atan2}(s_{4-3}, c_{4-3}) + x_3, \end{cases} \quad (3.14)$$

gdzie  $c_{4-3} = \frac{w_1^2 + w_2^2 - a_2^2 - a_3^2}{2a_2 a_3}$ ,  $s_{4-3} = \pm \sqrt{1 - c_{4-3}^2}$ . Równania (3.12), (3.13) i (3.14) dają łącznie następujące wyrażenie symboliczne dla kinematyki odwrotnej manipulatora IRb-6 zamontowanego na torze jezdnym

$$\mathbf{x} = \mathbf{r}(\mathbf{y}) = \begin{pmatrix} y_3 + y_1 \tan y_4 \\ -y_4 \\ \operatorname{atan2}(w_2, w_1) - \operatorname{atan2}(a_3 s_{4-3}, a_2 + a_3 c_{4-3}) \\ \operatorname{atan2}(s_{4-3}, c_{4-3}) + x_3 \\ \pi - y_5 \\ \pi - y_6 \end{pmatrix}, \quad (3.15)$$

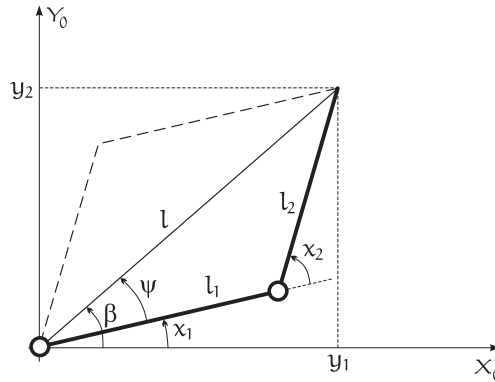
przy oznaczeniach  $c_{4-3}$ ,  $s_{4-3}$ ,  $w_1$  i  $w_2$  objaśnionych powyżej. Ze względu na ograniczenia zakresu ruchu w przegubach manipulatora, z dwóch możliwych rozwiązań (dostępnych ze względu na znak  $\pm$  w wyrażeniu na  $s_{4-3}$ ) należy wybrać rozwiązanie ze znakiem minus. Rozwiązanie (3.15) jest dobrze określone w całej przestrzeni roboczej manipulatora, poza położeniami dla których  $\cos y_4 = 0$ , co ma miejsce, gdy manipulator przyjmuje konfiguracje osobliwe. Sposób rozwiązania zadania odwrotnego w przypadku osobliwym objaśnimy w przykładzie 3.2.6. ■

### 3.1.2 Podejście geometryczne

Przykładem odmiennego od bezpośredniego podejścia algebraicznego sposobu podejścia do odwrotnego zadania kinematyki manipulatora jest podejście geometryczne. Cechę wyróżniającą podejście geometryczne stanowi dekompozycja przestrzennej geometrii manipulatora na szereg figur geometrii płaskiej. Dla zilustrowania podejścia geometrycznego posłużymy się dwoma przykładami.

<sup>†</sup>Porównaj z (2.68).

<sup>‡</sup>Zobacz przykład 3.1.1.



**Rysunek 3.1** Płaskie zależności geometryczne w manipulatorze typu podwójne wahadło.

### Przykład 3.1.3 (Manipulator typu podwójne wahadło)

Rozważmy najpierw manipulator typu podwójne wahadło z kinematyką opisaną wzorem (2.68). Na rysunku 3.1 pokazano trójkąt utworzony przez ramiona wahadła o długościach  $l_1$  i  $l_2$  oraz odcinek o długości  $l$  łączący początek układu podstawowego z efekтором manipulatora. Linia przerywaną zaznaczono inną możliwą konfigurację manipulatora prowadzącą do tego samego położenia efektora. Korzystając z równości  $l^2 = y_1^2 + y_2^2$  oraz  $\cos(\pi - x_2) = -\cos x_2$ , z twierdzenia cosinusów obliczamy

$$c_2 = \frac{y_1^2 + y_2^2 - l_1^2 - l_2^2}{2l_1 l_2}. \quad (3.16)$$

Zauważmy, że rozwiązanie równania (3.16) istnieje tylko wtedy, gdy jego prawa strona ma wartość z przedziału  $[-1, 1]$ . Ponieważ  $s_2 = \pm\sqrt{1 - c_2^2}$ , obliczamy

$$x_2 = \text{atan2}(s_2, c_2). \quad (3.17)$$

Ze wzoru (3.17) otrzymujemy obie wartości kąta  $x_2$  stanowiące rozwiązanie zadania.

W celu wyznaczenia kąta  $x_1$  wykorzystamy kąty  $\psi$  i  $\beta$ , zaznaczone na rysunku 3.1. Przy danym położeniu efektora manipulatora  $(y_1, y_2)$ , wyliczamy natychmiast

$$\beta = \text{atan2}(y_2, y_1).$$

Kąt  $\psi$  znajdujemy stosując ponownie twierdzenie cosinusów, które po podstawieniu  $l^2 = y_1^2 + y_2^2$  prowadzi do równania

$$\cos \psi = \frac{y_1^2 + y_2^2 + l_1^2 - l_2^2}{2l_1 \sqrt{y_1^2 + y_2^2}}.$$

Po wyznaczeniu kątów  $\beta$  i  $\psi$ , kąt  $\alpha_1$  obliczamy jako

$$\alpha_1 = \beta \pm \psi, \quad (3.18)$$

gdzie znak plus dotyczy przypadku  $\alpha_2 \leq 0$  (dokładniej  $0 \leq |\alpha_2| \bmod 2\pi < \pi$ ), a znak minus odnosi się do przypadku  $\alpha_2 > 0$  ( $\pi \leq |\alpha_2| \bmod 2\pi < 2\pi$ ). Zależności (3.18) i (3.17) są rozwiązaniem odwrotnego zadania kinematyki dla manipulatora typu podwójne wahadło. ■

### Przykład 3.1.4 (Manipulator typu SCARA)

Obecnie zastosujemy podejście geometryczne do rozwiązania odwrotnego zadania kinematyki dla manipulatora przemysłowego SCARA traktowanego jako manipulator redundantny z kinematyką opisaną wzorem (2.73).

Zauważmy, że po zrzutowaniu szkieletu manipulatora SCARA na płaszczyznę  $X_0Y_0$ , jego geometria w pełni odpowiada podwójnemu wahadłu pokazanemu na rysunku 3.1. W związku z tym, dwie pierwsze zmienne przegubowe ( $\alpha_1$  i  $\alpha_2$ ) mogą zostać wyliczone w sposób analogiczny jak w poprzednim przykładzie.

Z kolei, analiza możliwości ruchowych manipulatora wzdłuż osi  $Z_0$  prowadzi do wykrycia zależności

$$d_1 + \alpha_3 = y_3,$$

z której można wyliczyć zmienną przegubową  $\alpha_3$ . Ponieważ zmienna przegubowa  $\alpha_4$  nie wpływa na położenie efektoru manipulatora, a jedynie na jego orientację, jej wartość może być dowolna. ■

## 3.1.3 Metody jacobianowe

### Metoda jacobianu odwrotnego

Załóżmy, że ciągle zadanie odwrotne kinematyki we współrzędnych posiada rozwiązanie  $\mathbf{x}_d(t)$ . Wówczas, poprzez różniczkowanie względem czasu obu stron zależności (3.2), otrzymujemy układ równań różniczkowych

$$\dot{\mathbf{y}}_d = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}_d) \dot{\mathbf{x}}_d = \mathbf{J}^a(\mathbf{x}_d) \dot{\mathbf{x}}_d. \quad (3.19)$$

Przyjmijmy, że wymiar przestrzeni przegubowej jest równy wymiarowi przestrzeni zadaniowej ( $= n$ ) i niech wzdłuż trajektorii  $\mathbf{x}_d(t)$  jacobian analityczny  $\mathbf{J}^a(\mathbf{x}_d)$  będzie nieosobliwy. Algorytm *metody jacobianu odwrotnego* rozwiązywania odwrotnego zadania kinematyki polega na scałkowaniu układu nieautonomicznych równań różniczkowych

$$\dot{\mathbf{x}}_d = (\mathbf{J}^a)^{-1}(\mathbf{x}_d)\dot{\mathbf{y}}_d, \quad (3.20)$$

którego rozwiązanie  $\mathbf{x}_d(t)$  jest poszukiwaną trajektorią w przestrzeni przegubowej. Do wyznaczenia tej trajektorii potrzebna jest znajomość warunku początkowego  $\mathbf{x}_0$ , który wyliczamy rozwiązując punktowe zadanie odwrotne  $\mathbf{y}_0 = \mathbf{y}_d(0) = \mathbf{k}(\mathbf{x}_0)$ . Zadanie punktowe można rozwiązać korzystając z *algorytmu Newtona* pozwalającego wyznaczyć  $\mathbf{x}_0$  jako granicę  $\lim_{t \rightarrow +\infty} \boldsymbol{\xi}(t)$  trajektorii układu

$$\dot{\boldsymbol{\xi}} = -\alpha(\mathbf{J}^a)^{-1}(\boldsymbol{\xi})(\mathbf{k}(\boldsymbol{\xi}) - \mathbf{y}_0) \quad (3.21)$$

zainicjowanego w dowolnym stanie początkowym  $\boldsymbol{\xi}_0 \in \mathbb{R}^n$ , z parametrem  $\alpha > 0$  określającym szybkość zbieżności algorytmu. Z połączenia algorytmów (3.20) i (3.21) można otrzymać *asymptotyczny algorytm kinematyki odwrotnej*

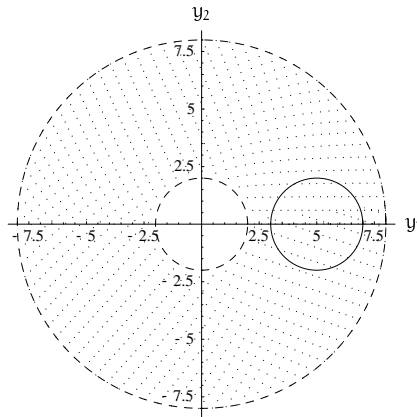
$$\dot{\mathbf{x}} = (\mathbf{J}^a)^{-1}(\mathbf{x})(\dot{\mathbf{y}}_d - \alpha(\mathbf{k}(\mathbf{x}) - \mathbf{y}_d)). \quad (3.22)$$

Przy dowolnym warunku początkowym trajektoria układu (3.22) dąży asymptotycznie do  $\mathbf{x}_d(t)$ .

Obecnie przejdziemy do zilustrowania metody jacobianu odwrotnego dwoma przykładami odwrotnego zadania kinematyki dla manipulatorów nieredundantnych.

### Przykład 3.1.5 (Manipulator typu podwójne wahadło)

Dla manipulatora typu podwójne wahadło, w przykładach 2.3.7 i 2.3.12, wyprowadziliśmy równania kinematyki we współrzędnych i jacobian analityczny w dwóch aspektach: współrzędnych  $X_0 Y_0 \varphi$  oraz współrzędnych  $X_0 Y_0$ . Ponieważ w pierwszym przypadku manipulator jest zawsze osobliwy, zajmujemy się rozwiązaniem odwrotnego zadania kinematyki dla przypadku drugiego, tzn. dla manipulatora z kinematyką opisaną równaniem (2.68) i jacobianem analitycznym danym zależnością (2.82). W obliczeniach przyjmujemy  $l_1 = 5$ ,  $l_2 = 3$ .



**Rysunek 3.2** Ścieżka ruchu zadana w przestrzeni roboczej manipulatora typu podwójne wahadło.

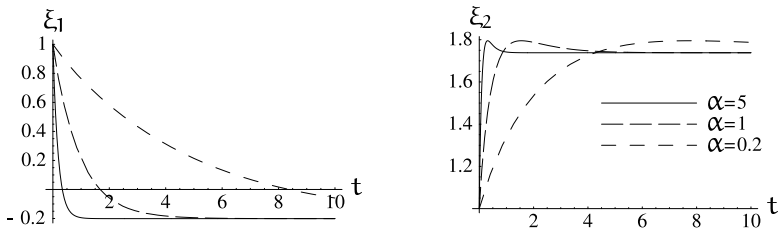
Założmy, że efektor manipulatora ma podążać wzdłuż trajektorii zadanej mającej postać okręgu

$$\mathbf{y}_d(t) = \begin{pmatrix} y_{d1}(t) \\ y_{d2}(t) \end{pmatrix} = \begin{pmatrix} 5 + 2 \sin(\frac{\pi}{5}t) \\ 2 \cos(\frac{\pi}{5}t) \end{pmatrix}, \quad (3.23)$$

w przedziale czasu  $\mathcal{I} = [0, 10]$ . Rysunek 3.2 przedstawia ścieżkę w przestrzeni roboczej manipulatora odpowiadającą zadanej trajektorii. Interesuje nas znalezienie trajektorii przegubowej manipulatora pozwalającej na realizację trajektorii (3.23).

Rozpoczynamy od rozwiązania punktowego zadania odwrotnego dla położenia  $\mathbf{y}_0 = \mathbf{y}_d(0) = (5, 2)^\top$ . W tym celu posłużymy się algorytmem Newtona opisanym przez równanie różniczkowe (3.21), z jacobianem  $\mathbf{J}^\alpha(\mathbf{x})$  i kinematyką  $\mathbf{k}(\mathbf{x})$  danymi odpowiednio przez (2.82) i (2.68). Układ równań (3.21) został zainicjowany w stanie początkowym  $\boldsymbol{\xi}_0 = (1, 1)^\top$ . Na rysunku 3.3 przedstawiono przebieg uzyskanych trajektorii dla różnych wartości parametru  $\alpha$  algorytmu. Jak widać, im większa wartość parametru  $\alpha$ , tym szybciej trajektoria układu dąży do wartości ustalonej, która jest warunkiem początkowym, pozwalającym na rozwiązanie odwrotnego zadania kinematyki przy pomocy algorytmu jacobianu odwrotnego. Z przeprowadzonych obliczeń wynika, że warunek początkowy  $\mathbf{x}_0 = (-0.201, 1.74)^\top$ \*

\*Zauważmy, że jest to jedno z dwóch możliwych rozwiązań zadania punktowego. Drugim rozwiązaniem jest  $(0.962, -1.74)^\top$ . To, które z rozwiązań otrzymamy, zależy od warunku początkowego w algorytmie Newtona.

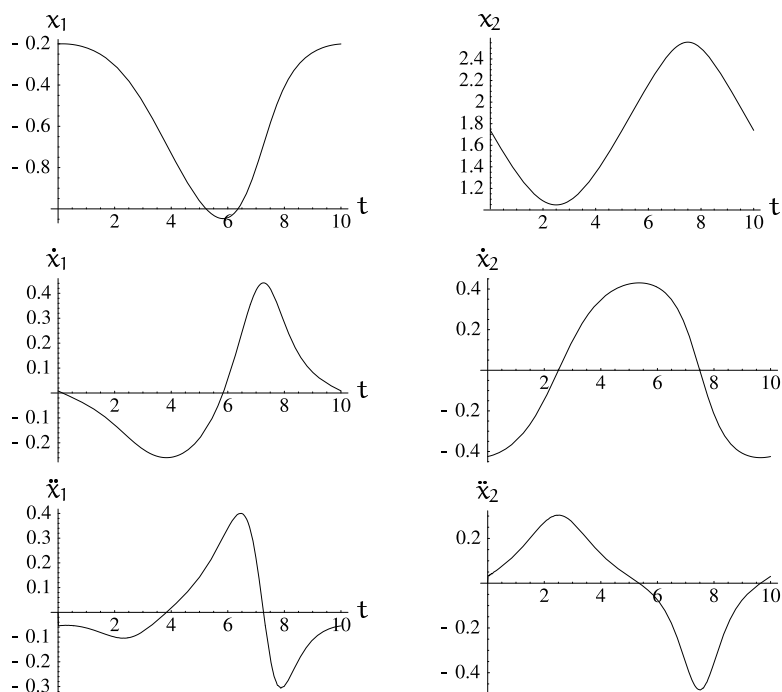


**Rysunek 3.3** Rozwiązanie punktowego zadania odwrotnego uzyskane przy pomocy algorytmu Newtona.

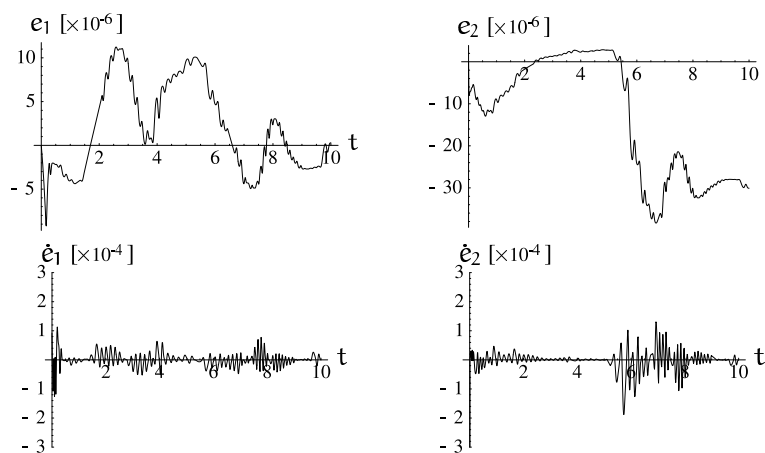
Aby rozwiązać zadanie odwrotne, całkujemy układ równań (3.20) poczynając od  $\mathbf{x}_0$ . Rysunek 3.4 przedstawia położenia, prędkości i przyspieszenia w przestrzeni przegubowej pozwalające na zrealizowanie trajektorii (3.23). Zauważmy, że po zakończeniu realizacji trajektorii, w chwili  $t = 10$ , manipulator wraca do stanu początkowego, co oznacza, że znaleziona trajektoria przegubowa może być wielokrotnie powtarzana. Jednak przy interpretacji wyników uzyskanych przy pomocy metod numerycznych powinniśmy zachować należyłą ostrożność. Bowiern, jeśli przyjrzymy się dokładnie przebiegowi błędów towarzyszących metodzie (zobacz rysunek 3.5, błędy zdefiniowano jako  $\mathbf{e}(t) = \mathbf{y}_d(t) - \mathbf{k}(\mathbf{x}(t))$ ,  $\dot{\mathbf{e}} = \dot{\mathbf{y}}_d(t) - \mathbf{J}^a(\mathbf{x}(t))\dot{\mathbf{x}}(t)$ , gdzie  $\mathbf{x}(t)$  i  $\dot{\mathbf{x}}(t)$  oznaczają położenia i prędkości w przegubach manipulatora uzyskane jako rozwiązanie) zauważymy, że po jednokrotnej realizacji trajektorii pojawił się błąd położenia efektor manipulatora. Jest to spowodowane niedokładnością metod numerycznych zastosowanych do rozwiązania równań różniczkowych<sup>†</sup>. Przy powtarzaniu trajektorii powstające błędy mogą się kumulować.

Użyjemy teraz do rozwiązania zadania odwrotnego asymptotycznego algorytmu kinematyki odwrotnej opisanego zależnością (3.22). Podobnie jak w przypadku algorytmu Newtona, układ (3.22) został zainicjowany w punkcie  $\mathbf{x}_0 = (1, 1)^T$ . Rysunek 3.6 przedstawia trajektorie uzyskane przy pomocy algorytmu asymptotycznego z parametrem  $\alpha = 1$ . Błędy położenia i prędkości efektor, zdefiniowane jak poprzednio, zostały pokazane na rysunku 3.7. Jak widać, błędy, których głównym źródłem w tym przypadku jest metoda, maleją asymptotycznie do zera. Widać także, że praktycznie od chwili  $t = 4$  uzyskana trajektoria pokrywa się z trajektorią wy-

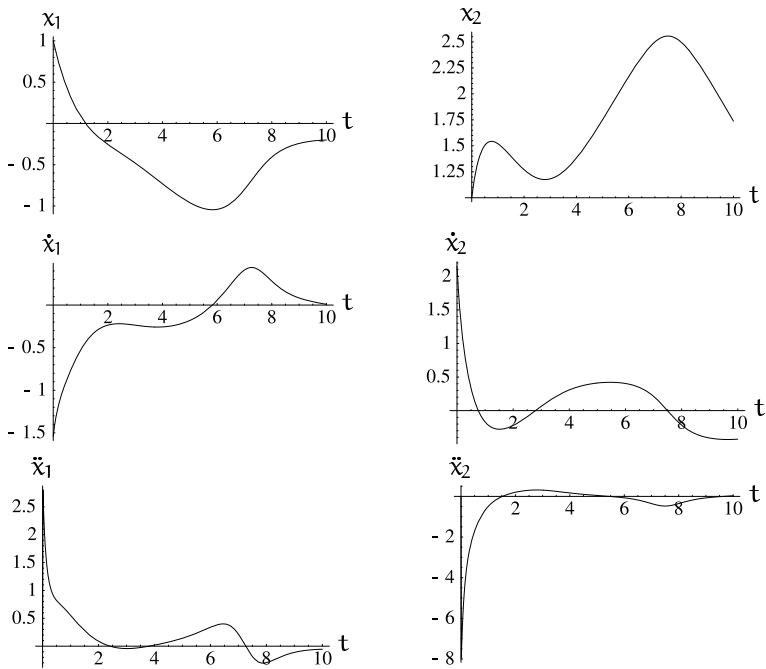
<sup>†</sup>W przykładach prezentowanych w tym rozdziale symulacje wykonano w środowisku MATHEMATICA<sup>®</sup>, a do numerycznego rozwiązywania równań różniczkowych zastosowano metodę Rungego-Kutty czwartego lub piątego rzędu.



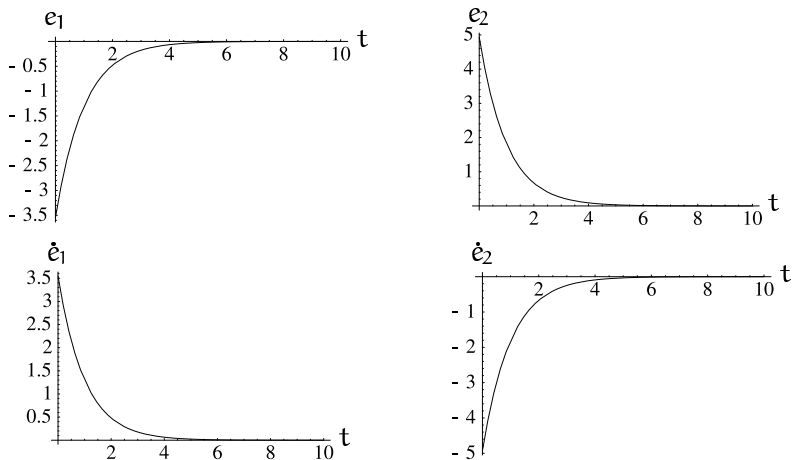
**Rysunek 3.4** Rozwiązanie zadania odwrotnego dla podwójnego wahadła uzyskane przy pomocy algorytmu jacobianu odwrotnego.



**Rysunek 3.5** Błędy położenia i prędkości efektora przy algorytmie jacobianu odwrotnego.

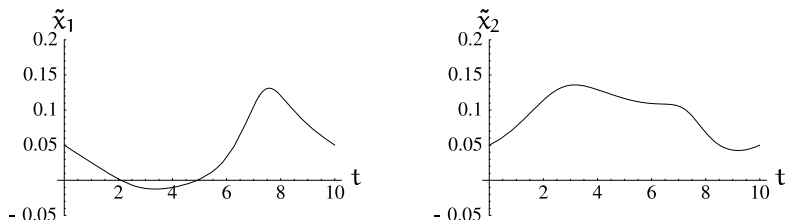


**Rysunek 3.6** Rozwiązanie zadania odwrotnego dla podwójnego wahadła uzyskane przy zastosowaniu asymptotycznego algorytmu kinematyki odwrotnej.



**Rysunek 3.7** Błędy położenia i prędkości efektora przy asymptotycznym algorytmie kinematyki odwrotnej.





**Rysunek 3.8** Błędy położenia przegubów manipulatora wynikające z zastosowania algorytmu jacobianu odwrotnego przy niedokładności wyznaczenia stanu początkowego  $(\tilde{x}_1(0), \tilde{x}_2(0)) = (0.05, 0.05)$ .

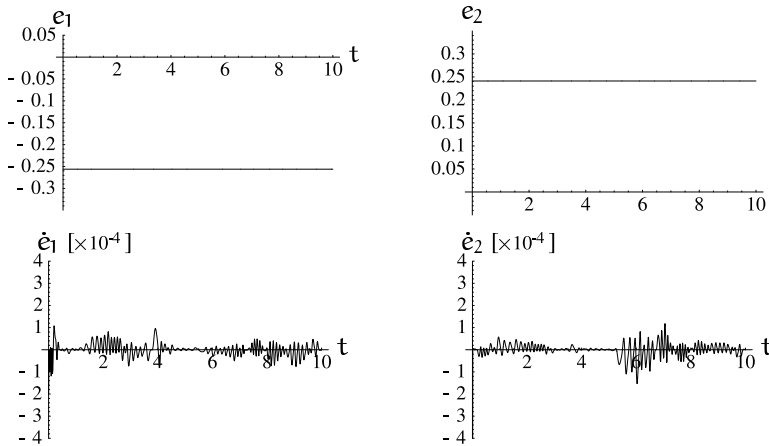
znaczoną za pośrednictwem algorytmu jacobianu odwrotnego, przedstawioną na rysunku 3.4. Oczywiście, i tym razem pojawiają się błędy numeryczne, ale wielokrotnie mniejsze od błędów metody. Szybkość zbieżności błędu zależy od wartości parametru  $\alpha$ , jednakże dużym wartościom  $\alpha$ , dla których błędy szybko dążą do zera, towarzyszą duże prędkości i przyspieszenia w początkowym okresie realizacji trajektorii.

Na zakończenie zbadamy wpływ dokładności wyznaczenia warunku początkowego na przebieg trajektorii uzyskanej jako rozwiązanie odwrotnego zadania kinematyki przy użyciu algorytmu jacobianu odwrotnego. Przyjmijmy, że przy wyznaczaniu stanu początkowego każdego z przegubów popełniono błąd o wartości 0.05. Rysunek 3.8 pokazuje, jak błąd stanu początkowego wpływa na odchylenie  $\tilde{\mathbf{x}}$  przebiegu trajektorii od trajektorii pokazanej na rysunku 3.4. Rysunek 3.9 przedstawia błędy położenia i prędkości efektor manipulatora towarzyszące takiemu odchyleniu. Z rysunku wynika, że dla rozwiązania uzyskanego w wyniku zastosowania algorytmu jacobianu odwrotnego błąd prędkości efektor manipulatora nie zależy od dokładności wyznaczenia warunków początkowych, występuje natomiast błąd położenia efektor zależny od niedokładności wyznaczenia tych warunków. ■

### Przykład 3.1.6 (Manipulator IRb-6 na torze jezdnym)

Założmy, że manipulator IRb-6 zamontowany na torze jezdnym, z kinematyką zdefiniowaną wzorem (2.76), ma za zadanie przemieszczać efektor wzdłuż linii śrubowej

$$\begin{pmatrix} y_{d1}(t) \\ y_{d2}(t) \\ y_{d3}(t) \end{pmatrix} = \begin{pmatrix} 0.7 + 0.2 \sin\left(\frac{\pi}{2}t\right) \\ -0.7 - 0.01t \\ 0.7 + 0.2 \cos\left(\frac{\pi}{2}t\right) \end{pmatrix}, \quad (3.24)$$

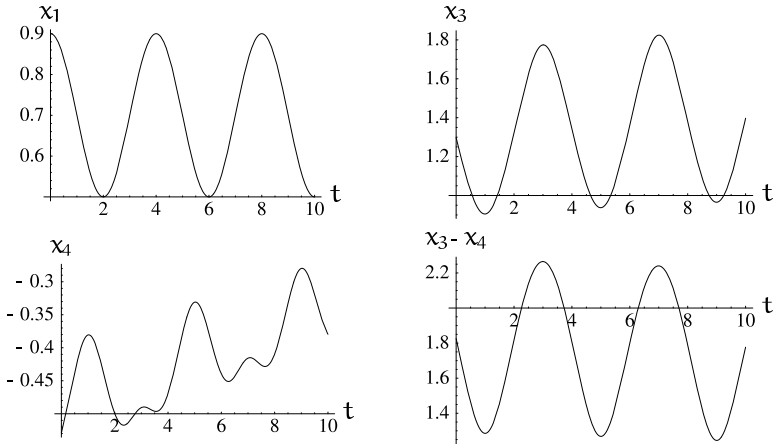


**Rysunek 3.9** Błędy położenia i prędkości efektora powstałe przy zastosowaniu algorytmu jacobianu odwrotnego przy niedokładności wyznaczenia stanu początkowego  $(\tilde{x}_1(0), \tilde{x}_2(0)) = (0.05, 0.05)$ .

przy nieziennej orientacji efektora danej przez

$$\begin{pmatrix} y_{d4}(t) \\ y_{d5}(t) \\ y_{d6}(t) \end{pmatrix} = \begin{pmatrix} 0 \\ \pi \\ \pi \end{pmatrix}, \quad (3.25)$$

w przedziale  $\mathcal{I} = [0, 10][s]$ . W celu określenia przebiegu trajektorii przegubowej manipulatora rozwiązaliśmy układ równań różniczkowych (3.20) z jacobianem analitycznym  $J^a(\mathbf{x})$  postaci (2.87), zainicjowany w punkcie początkowym  $\mathbf{x}(0) = (0.9, 0., 1.29, -0.529, 0., 0.)^T$  (wartość uzyskana przy użyciu algorytmu Newtona (3.21)). Rysunek 3.10 przedstawia przebiegi czasowe zmiennych przegubowych  $x_1, x_3, x_4$  oraz  $x_3 - x_4$  uzyskane w wyniku obliczeń. Zmienne  $x_2, x_5$  i  $x_6$ , nie pokazane na rysunku, miały stałe wartości równe 0. Jak widać, uzyskana trajektoria znajduje się w zbiorze konfiguracji dopuszczalnych  $\mathcal{X}$  określonym przez (2.78), co oznacza, że może zostać zrealizowana. Chcemy zwrócić uwagę Czytelnika na fakt, że częstokroć takie wyznaczenie trajektorii zadanej w przestrzeni zadaniowej manipulatora, aby znalazła się ona wewnątrz jego przestrzeni roboczej, a tym samym, by zachodziła wymieniona własność, jest trudne. Główną tego przyczyną są problemy z analitycznym określeniem granic przestrzeni roboczej manipulatora, zwłaszcza gdy współrzędne zadaniowe zawierają część określającą orientację efektoru. Zadanie można uprościć przez zmniejszenie wymiaru przestrzeni zadaniowej manipulatora i potraktowanie go jako redundantny.



**Rysunek 3.10** Rozwiązanie zadania odwrotnego dla manipulatora IRb-6 zamontowanego na torze jezdny uzyskane przy pomocy algorytmu jacobianu odwrotnego.

W przytoczonym przykładzie można np. zrezygnować z określenia orientacji efektora manipulatora i przyjąć, że manipulator IRb-6 zamontowany na torze jezdny jest manipulatorem redundantnym z kinematyką opisaną równaniami (2.77)<sup>‡</sup>. ■

### Metoda jacobianu pseudoodwrotnego

Weźmy teraz pod uwagę odwrotne zadanie kinematyki dla manipulatora redundantnego, którego kinematyka jest opisana formułą  $\mathbf{y} = \mathbf{k}(\mathbf{x})$  przy warunku  $\dim \mathbf{x} = n > \dim \mathbf{y} = m$ . Jeżeli istnieje rozwiązanie  $\mathbf{x}_d(t)$  zadania, równanie (3.19) obowiązuje. Niech jacobian analityczny  $\mathbf{J}^a(\mathbf{x})$  ma wzdłuż trajektorii  $\mathbf{x}_d(t)$  pełny rząd (=  $m$ ). Ponieważ zwykła odwrotność macierzy  $\mathbf{J}^a(\mathbf{x})$  nie istnieje, w układzie (3.20) trzeba użyć (prawostronnej) pseudoodwrotności jacobianu analitycznego<sup>§</sup>. Przypominamy, że pseudoodwrotność macierzy  $\mathbf{A}$  opisuje rozwiązanie układu równań liniowych

$$\mathbf{w} = \mathbf{A}\mathbf{v}$$

spełniające warunek minimalizacji formy kwadratowej

$$V(\mathbf{v}) = \frac{1}{2} \mathbf{v}^T \mathbf{W} \mathbf{v}$$

<sup>‡</sup>Zobacz przykłady 3.1.8–3.1.11.

<sup>§</sup>Zobacz dodatek A.1.

z symetryczną, dodatnio określoną macierzą  $\mathbf{W}$ . Wyznaczenie pseudoodwrotności sprowadza się zatem do rozwiązania zadania optymalizacji kwadratowej z ograniczeniami równościowymi. Po rozwiązaniu tego zadania dla  $\mathbf{A} = \mathbf{J}^a(\mathbf{x})$  otrzymujemy następującą zależność

$$\mathbf{J}_W^{a\#}(\mathbf{x}) = \mathbf{W}^{-1} \mathbf{J}^{aT}(\mathbf{x}) (\mathbf{J}^a(\mathbf{x}) \mathbf{W}^{-1} \mathbf{J}^{aT}(\mathbf{x}))^{-1}. \quad (3.26)$$

Macierz (3.26), zwana *pseudoodwrotnością* jakobianu analitycznego jest macierzą rozmiaru  $n \times m$  o tej własności, że

$$\mathbf{J}^a(\mathbf{x}) \mathbf{J}_W^{a\#}(\mathbf{x}) = \mathbb{I}_m. \quad (3.27)$$

Pseudoodwrotność jakobianu jest podstawowym składnikiem *metody jakobianu pseudoodwrotnego*, według której rozwiązanie zadania odwrotnego otrzymuje się przez scałkowanie (nieautonomicznego) układu równań różniczkowych

$$\dot{\mathbf{x}}_d = \mathbf{J}_W^{a\#}(\mathbf{x}_d) \dot{\mathbf{y}}_d, \quad (3.28)$$

zainicjowanego w punkcie  $\mathbf{x}_0$ , takim że  $\mathbf{y}_0 = \mathbf{y}_d(0) = \mathbf{k}(\mathbf{x}_0)$ . Analogicznie do zależności (3.21), warunek początkowy  $\mathbf{x}_0$  można obliczyć przy pomocy algorytmu Newtona, biorąc granicę  $\lim_{t \rightarrow +\infty} \boldsymbol{\xi}(t)$  trajektorii układu

$$\dot{\boldsymbol{\xi}} = -\alpha \mathbf{J}_W^{a\#}(\boldsymbol{\xi}) (\mathbf{k}(\boldsymbol{\xi}) - \mathbf{y}_0). \quad (3.29)$$

Odpowiednikiem algorytmu (3.22) jest w przypadku redundantnym następujący *algorytm asymptotyczny*

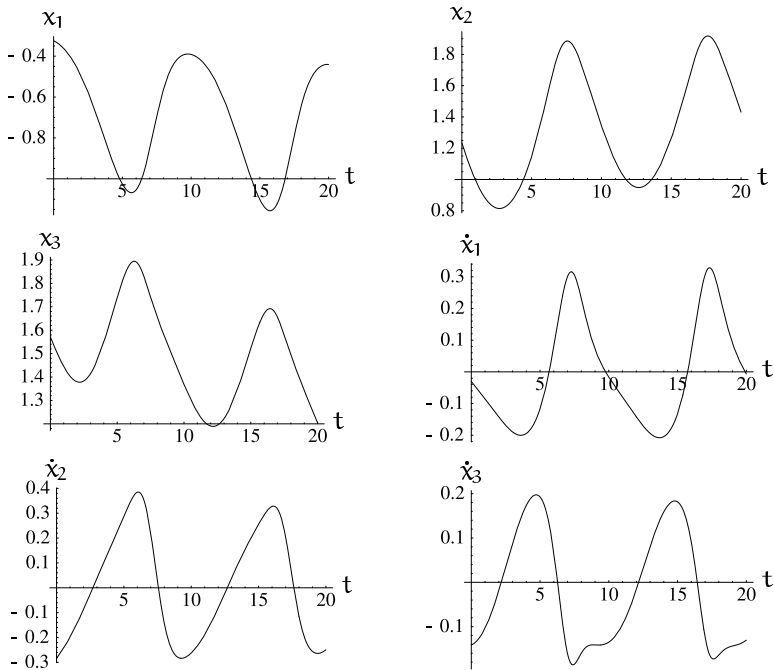
$$\dot{\mathbf{x}} = \mathbf{J}_W^{a\#}(\mathbf{x}) (\dot{\mathbf{y}}_d - \alpha (\mathbf{k}(\mathbf{x}) - \mathbf{y}_d)). \quad (3.30)$$

Poniżej, na przykładach manipulatora typu potrójne wahadło i manipulatora IRb-6 zamontowanego na torze jezdny, przedstawimy rozwiązanie odwrotnego zadania kinematyki przy zastosowaniu pseudoodwrotności jakobianu  $\mathbf{J}_W^{a\#}(\mathbf{x})$  z macierzą  $\mathbf{W} = \mathbb{I}_n$ .

### Przykład 3.1.7 (Manipulator typu potrójne wahadło)

Rozpatrzmy manipulator redundantny typu potrójne wahadło z kinematyką opisaną wzorem (2.71) i jakobianem analitycznym (2.85). Przyjmijmy długości ramion manipulatora równe  $l_1 = 5$ ,  $l_2 = 3$ ,  $l_3 = 2$ . Zadaniem manipulatora jest wykonanie ruchu wzdłuż trajektorii w postaci okręgu o równaniach

$$\mathbf{y}_d(t) = \begin{pmatrix} y_{d1}(t) \\ y_{d2}(t) \end{pmatrix} = \begin{pmatrix} 5 + 2 \sin(\frac{\pi}{5}t) \\ 2 \cos(\frac{\pi}{5}t) \end{pmatrix}, \quad (3.31)$$

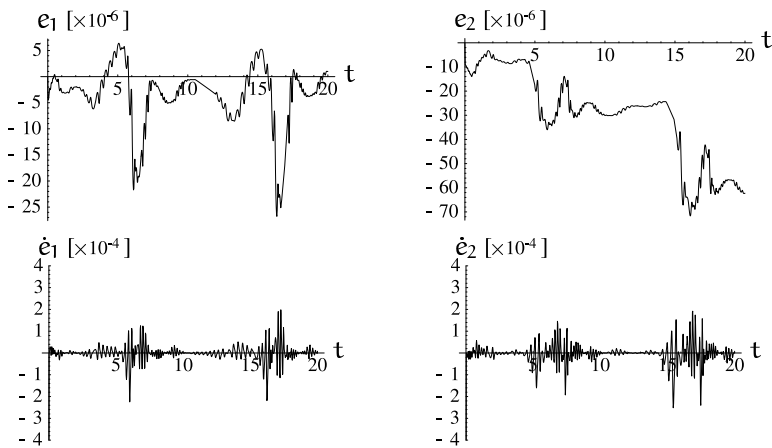


**Rysunek 3.11** Rozwiązanie zadania odwrotnego dla potrójnego wahadła uzyskane przy pomocy algorytmu jacobianu pseudoodwrotnego.

w przedziale czasu  $\mathcal{I} = [0, 20]$ . Zwróćmy uwagę na fakt, że podczas realizacji trajektorii efektor manipulatora powinien wykonać dwa obiegi okręgu w przestrzeni roboczej.

Po scałkowaniu układu równań różniczkowych (3.29), zainicjowanego w punkcie  $\xi = (1, 1, 1)^T$ , otrzymujemy rozwiązanie  $\mathbf{x}_0 = (-0.325, 1.23, 1.57)^T$  punktowego zadania odwrotnego dla  $\mathbf{y}_{d0} = \mathbf{y}_d(0) = (5, 2)^T$ <sup>¶</sup>. Punkt  $\mathbf{x}_0$  posłuży jako warunek początkowy przy rozwiązaniu układu równań (3.28), pozwalających na znalezienie rozwiązania postawionego zadania. Trajektorja uzyskana przez scałkowanie układu (3.28) została przedstawiona na rysunku 3.11. Zauważmy, że kolejnym obiegiem efektora po okręgu towarzyszą różne sekwencje położeń przegubów manipulatora. Przy wielokrotnym powtarzaniu trajektorii daje się zauważyć tendencja do poruszania przede wszystkim początkowymi przegubami manipulatora. Analiza przebiegu błędów położenia i prędkości efektora pokazanych na rysunku 3.12 wskazuje, że, podobnie jak w przypadku metody jacobianu odwrotnego (zo-

<sup>¶</sup>Zobacz przykład 3.1.12.



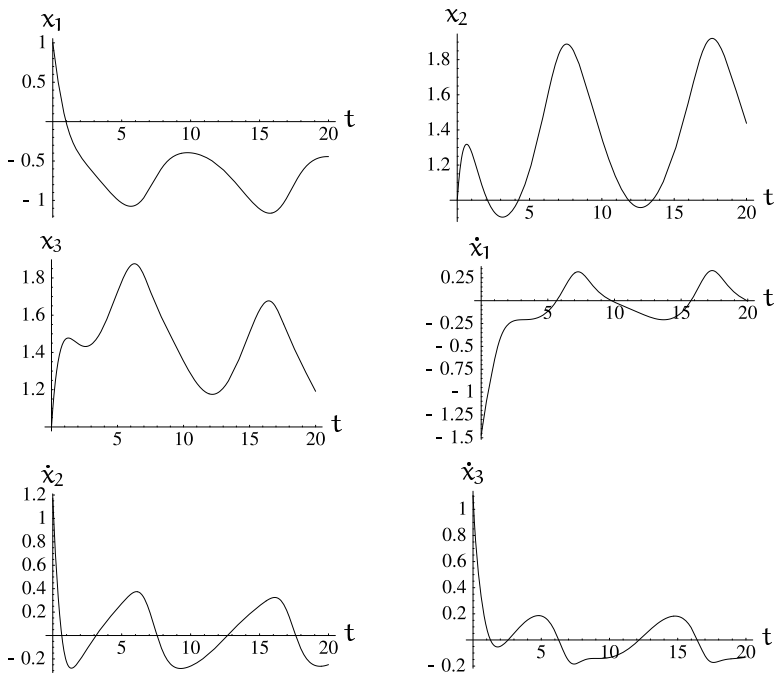
**Rysunek 3.12** Błędy położenia i prędkości efektora przy algorytmie jacobianu pseudoodwrotnego.

bacz rysunek 3.5), są to błędy numeryczne.

Obecnie, do rozwiązania zadania użyjemy algorytmu asymptotycznego (3.30), wynikającego z połączenia algorytmu jacobianu pseudoodwrotnego i algorytmu Newtona. Rozwiązanie otrzymane w wyniku scałkowania równań (3.30) przy parametrze  $\alpha = 5$  przedstawia rysunek 3.13. Podobnie jak w poprzednim przypadku, układ równań został zainicjowany w punkcie  $\mathbf{x} = (1, 1, 1)^T$ . Z rysunku widać, że uzyskane rozwiązanie dąży asymptotycznie do rozwiązania uzyskanego przy użyciu algorytmu jacobianu pseudoodwrotnego (zobacz rysunek 3.11), czemu w początkowej fazie ruchu towarzyszą duże prędkości w przegubach. Oczywiście, tak samo jak w przypadku nieredundantnym, prędkości te zależą od wartości parametru  $\alpha$ . ■

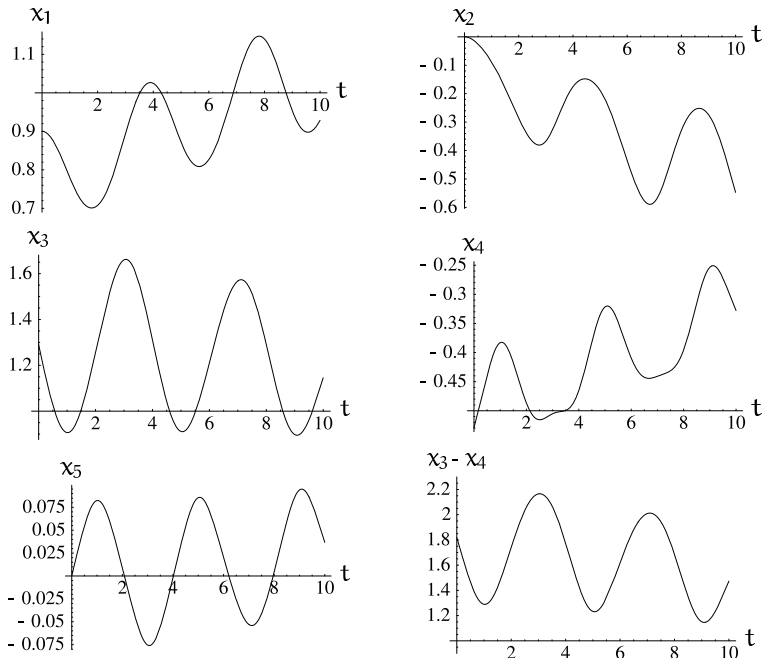
### Przykład 3.1.8 (Manipulator IRb-6 na torze jezdnym)

W tym przykładzie manipulator IRb-6 zamontowany na torze jezdny potraktujemy jako manipulator redundantny z kinematyką (2.77). Żądamy, aby efektor manipulatora poruszał się wzdłuż trajektorii  $\mathbf{y}_d(t)$  postaci (3.24) dla czasu  $t \in \mathcal{I} = [0, 10][s]$ . Jak widać, w tym przykładzie określiliśmy jedynie sposób zmian położenia efektor manipulatora, nie precyzując jak ma się zmieniać jego orientacja. Ponieważ rozpatrywany manipulator jest redundantny, do rozwiązania odwrotnego zadania kinematyki użyjemy metody jacobianu pseudoodwrotnego zdefiniowanej wzorem (3.28). Przebieg trajektorii przegubowej manipulatora dla konfiguracji początkowej  $\mathbf{x}(0) = (0.9, 0., 1.29, -0.529, 0., 0.)^T$  (wartość uzyskana za pomocą algorytmu New-



**Rysunek 3.13** Rozwiązanie zadania odwrotnego dla potrójnego wahadła uzyskane przy pomocy algorytmu asymptotycznego z pseudoodwrotnością jacobianu.

tona (3.29)) przedstawia rysunek 3.14. Nie pokazany na rysunku kąt  $x_6$  przyjmował stałe wartość 0. Dokonajmy krótkiego porównania tej trajektorii z trajektorią uzyskaną w przykładzie 3.1.6 w wyniku zastosowania metody jacobianu odwrotnego, pokazaną na rysunku 3.10. Oczywiście, również ta ostatnia trajektorja jest jednym z rozwiązań zadania odwrotnego. Jednakże, w przypadku nieredundantnym ustalenie orientacji, z jaką miał zostać wykonany ruch po linii śrubowej spowodowało, że ruch efektora został zrealizowany przez poruszanie jedynie pierwszego, trzeciego i czwartego przegubu manipulatora. W rozwiązaniu uzyskanym metodą jacobianu pseudoodwrotnego nie porusza się jedynie szósty przegub manipulatora. Dopiero po wielokrotnym powtórzeniu ruchu po linii śrubowej otrzymujemy tą metodą rozwiązanie, w którym ruch jest praktycznie efektem poruszania jedynie dwóch pierwszych przegubów manipulatora. Przykład takiego rozwiązania przedstawiono na rysunku 3.15. Rozwiązanie to uzyskano przez scałkowanie układu równań różniczkowych (3.28) zainicjowanych w punkcie  $\mathbf{x}(0) = (1.8, -0.91, 0.091, 0.076, 0.27, 0)^T$  wyliczonym przy pomocy algorytmu (3.29). Zauważmy, że otrzymana trajektorja nie leży w granicach



**Rysunek 3.14** Rozwiązanie zadania odwrotnego dla manipulatora IRb-6 zamontowanego na torze jezdnym uzyskane przy pomocy algorytmu jacobianu pseudo-odwrotnego.

przestrzeni przegubowej manipulatora opisanej zależnością (2.78), przez co jej realizacja nie jest możliwa. Uniknięcie takich sytuacji umożliwi przedstawiona niżej modyfikacja metody jacobianu pseudoodwrotnego, zilustrowana przykładem 3.1.9. ■

Okazuje się, że trajektoria  $\mathbf{x}_d(t)$  spełniająca układ równań (3.28) stanowi szczególne rozwiązanie odwrotnego zadania kinematyki dla manipulatora redundantnego. Jak bowiem łatwo zauważyć, zależność (3.27) prowadzi do spostrzeżenia

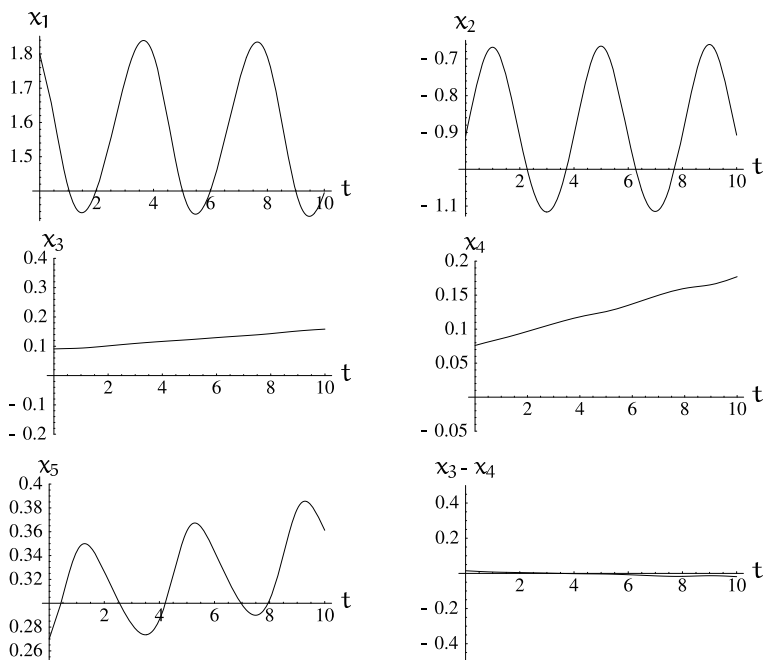
$$\mathbf{J}^a(\mathbf{x}) \left( \mathbb{I}_n - \mathbf{J}_W^{a\#}(\mathbf{x}) \mathbf{J}^a(\mathbf{x}) \right) = \mathbf{0},$$

z którego wynika, że odwzorowanie

$$\mathbf{v} \mapsto \left( \mathbb{I}_n - \mathbf{J}_W^{a\#}(\mathbf{x}) \mathbf{J}^a(\mathbf{x}) \right) \mathbf{v} \quad (3.32)$$

stanowi projekcję przestrzeni  $\mathbb{R}^n$  na przestrzeń zerową  $\text{Ker } \mathbf{J}^a(\mathbf{x})$  jacobianu





**Rysunek 3.15** Alternatywne rozwiązanie zadania odwrotnego dla manipulatora IRb-6 zamontowanego na torze jezdnyim uzyskane przy pomocy algorytmu jacobianu pseudoodwrotnego.

analitycznego. Korzystając z tej własności, możemy do prawej strony układu równań (3.28) dodać dowolny wektor należący do  $\text{Ker } \mathbf{J}^a(\mathbf{x})$ , będący obrazem pewnego wektora  $\mathbf{v} \in \mathbb{R}^n$  przez odwzorowanie (3.32). W efekcie, otrzymujemy *metodę jacobianu pseudoodwrotnego z projekcją*, zgodnie z którą rozwiązanie ogólne zadania odwrotnego dla manipulatora redundantnego spełnia następujący układ równań różniczkowych

$$\dot{\mathbf{x}}_d = \mathbf{J}_W^{a\#}(\mathbf{x}_d)\dot{\mathbf{y}}_d + \left( \mathbb{I}_n - \mathbf{J}_W^{a\#}(\mathbf{x}_d)\mathbf{J}^a(\mathbf{x}_d) \right) \mathbf{v}, \quad (3.33)$$

$\mathbf{v} \in \mathbb{R}^n$ , z warunkiem początkowym  $\mathbf{x}_0$  wyliczonym przy pomocy *algorytmu Newtona z projekcją*

$$\dot{\boldsymbol{\xi}} = -\alpha \mathbf{J}_W^{a\#}(\boldsymbol{\xi})(\mathbf{k}(\boldsymbol{\xi}) - \mathbf{y}_0) + \left( \mathbb{I}_n - \mathbf{J}_W^{a\#}(\boldsymbol{\xi})\mathbf{J}^a(\boldsymbol{\xi}) \right) \mathbf{v}. \quad (3.34)$$

Oba algorytmy (3.34), (3.33) można połączyć w jeden *algorytm asymptotyczny kinematyki odwrotnej*, dla którego warunek początkowy  $\mathbf{x}_0$  może

być wybrany w sposób arbitralny,

$$\dot{\mathbf{x}} = \mathbf{J}_W^{a\#}(\mathbf{x})(\dot{\mathbf{y}}_d - \alpha(\mathbf{k}(\mathbf{x}) - \mathbf{y}_d)) + \left(\mathbb{I}_n - \mathbf{J}_W^{a\#}(\mathbf{x})\mathbf{J}^a(\mathbf{x})\right)\mathbf{v}. \quad (3.35)$$

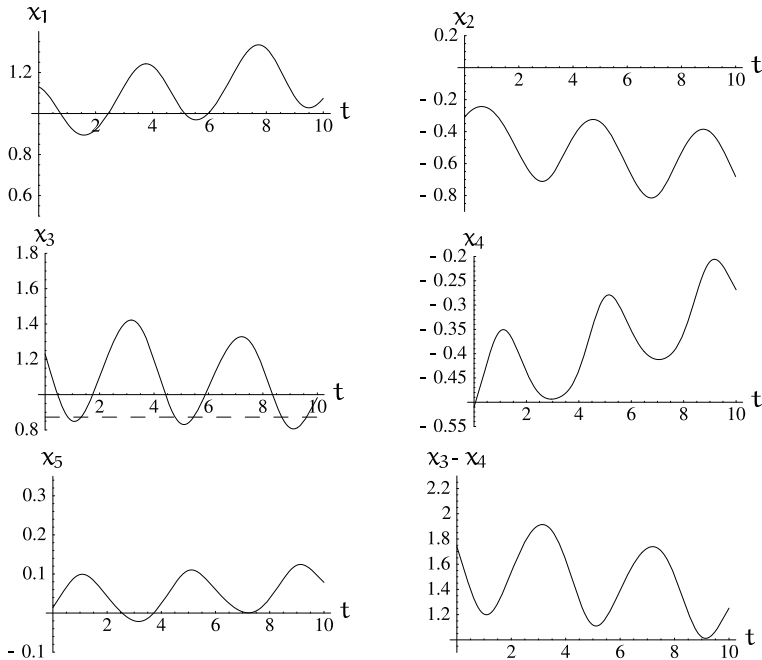
Przy  $t \rightarrow +\infty$  rozwiązanie układu (3.35) dąży do trajektorii układu (3.28). Wybór wektora  $\mathbf{v}$  zależy od dodatkowych kryteriów, jakie powinna spełniać trajektoria przegubowa  $\mathbf{x}(t)$ . Jeżeli przeguby manipulatora są ograniczone, naturalnym kryterium jest wymaganie, by położenia przyjmowane przez przeguby były możliwie bliskie środka zakresu ruchu przegubów (dalekie od mechanicznych ograniczeń ruchu przegubów). Funkcja odległości może mieć postać

$$d(\mathbf{x}) = \frac{1}{2n} \sum_{i=1}^n \left( \frac{x_i - \bar{x}_i}{x_{iM} - x_{iM}} \right)^2, \quad (3.36)$$

gdzie  $\bar{x}_i$ ,  $x_{iM}$ ,  $x_{im}$  oznaczają, odpowiednio, współrzędne środka zakresu ruchu oraz górnego i dolnego ograniczenia ruchu w  $i$ -tym przegubie manipulatora. Aby uzyskać rozwiązanie odwrotnego zadania kinematyki respektujące wymaganie unikania granicznych położeń przegubów manipulatora, w algorytmie (3.33) lub (3.35) należy zdefiniować wektor  $\mathbf{v} = -\beta \frac{\partial d(\mathbf{x})}{\partial \mathbf{x}}$ ,  $\beta > 0$ . Inne wybory wektora  $\mathbf{v}$  mogą mieć na celu maksymalizację manipulowalności manipulatora. Przedstawimy je w podrozdziale 3.1.5.

### Przykład 3.1.9 (Manipulator IRb-6 na torze jezdnym)

Założmy, że efektor manipulatora IRb-6 zamontowanego na torze jezdym, opisanego kinematyką (2.77), ma wykonywać ruch wzdłuż trajektorii  $\mathbf{y}_d(t)$  opisaną wzorem (3.24) dla chwil czasu  $t \in \mathcal{I} = [0, 10][s]$ . Interesuje nas znalezienie trajektorii przegubowej manipulatora odpowiadającej trajektorii określonej w przestrzeni zadaniowej. Jedno z rozwiązań takiego zadania, uzyskane w wyniku scałkowania układu (3.28) z macierzą  $\mathbf{W} = \mathbb{I}_6$ , przedstawiono na rysunku 3.16 (dwa inne rozwiązania można znaleźć w przykładzie 3.1.8). W przedstawionym przypadku jako punkt startowy wybraliśmy punkt  $\mathbf{x}(0) = (1.13, -0.31, 1.23, -0.51, 0.014, 0.)^T$  znaleziony w wyniku rozwiązania równań (3.29). Wartość nie pokazanego na rysunku 3.16 kąta  $x_6$  jest stale równa 0. Jak widać z rysunku, otrzymane rozwiązanie nie leży w dopuszczalnym zakresie zmian zmiennych przegubowych określonym przez (2.78) (linią przerywaną pokazano minimalną wartość kąta  $x_3$ ). Aby zapobiec tej sytuacji, należy w miejsce algorytmu (3.28) zastosować algorytm (3.33) z wektorem  $\mathbf{v}$  wybranym jako antygradient odległości (3.36). Rozwiązanie uzyskane w ten sposób (z macierzą  $\mathbf{W} = \mathbb{I}_6$ ) przedstawia rysunek 3.17. ■



**Rysunek 3.16** Rozwiązanie zadania odwrotnego dla manipulatora IRb-6 zamontowanego na torze jezdnyim uzyskane przy pomocy algorytmu jacobianu pseudo-odwrotnego.

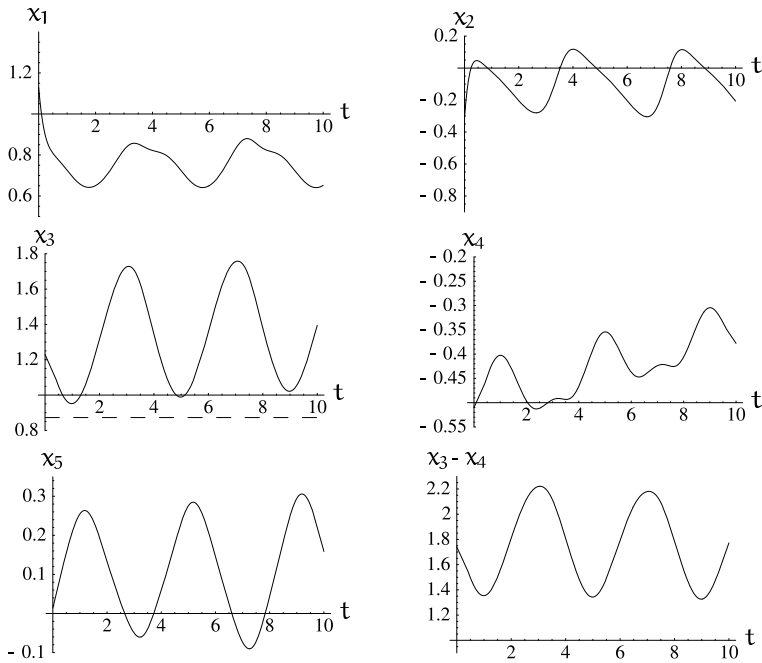
### Metoda jacobianu rozszerzonego

Problem, jaki się pojawia przy przejściu od układu (3.19) do układu (3.20) przy rozpatrywaniu kinematyki manipulatorów redundantnych, bywa rozwiązywany inaczej niż poprzez zastosowanie pseudoodwrotności jacobianu. Naturalną alternatywę stanowi wprowadzenie tzw. *rozszerzonego jacobianu analitycznego*, będącego macierzą kwadratową i nieosobliwą. Załóżmy, że dany jest jacobian analityczny  $\mathbf{J}^a(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x})$  rozmiaru  $m \times n$ . Wybierzmy zestaw  $n - m$  gładkich (analitycznych) funkcji  $\mathbf{f} = (f_1, f_2, \dots, f_{n-m})^T : \mathbb{R}^n \rightarrow \mathbb{R}^{n-m}$  i wprowadźmy  $n - m$  ograniczeń równościowych (typu ograniczeń konfiguracyjnych)

$$f_1(\mathbf{x}) = f_2(\mathbf{x}) = \dots = f_{n-m}(\mathbf{x}) = 0. \quad (3.37)$$

Żądamy, by dodatkowe ograniczenia były niezależne,

$$\text{rank} \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}) = n - m,$$



**Rysunek 3.17** Rozwiązanie zadania odwrotnego dla manipulatora IRb-6 zamontowanego na torze jezdnym uzyskane przy pomocy algorytmu jacobianu pseudo-odwrotnego z projekcją.

i tworzymy rozszerzony jacobian analityczny

$$\mathbf{J}^{ae}(\mathbf{x}) = \begin{bmatrix} \frac{\partial \mathbf{k}}{\partial \mathbf{x}} \\ \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \end{bmatrix}(\mathbf{x}), \quad (3.38)$$

który jest macierzą kwadratową. Ograniczenia  $\mathbf{f}(\mathbf{x})$  muszą być dobrane w taki sposób, by rozszerzony jacobian był nieosobliwy, przynajmniej wzdłuż poszukiwanej trajektorii  $\mathbf{x}_d(t)$ . Zakładając, że rozwiązanie zadania odwrotnego przy ograniczeniach (3.37) istnieje, zróżniczkujemy zależność

$$\begin{cases} \mathbf{y} = \mathbf{k}(\mathbf{x}) \\ \mathbf{0} = \mathbf{f}(\mathbf{x}) \end{cases}$$

względem czasu otrzymując

$$\begin{pmatrix} \dot{\mathbf{y}} \\ \mathbf{0} \end{pmatrix} = \mathbf{J}^{ae}(\mathbf{x})\dot{\mathbf{x}}. \quad (3.39)$$

Stosownie do *metody jacobianu rozszerzonego*, rozwiązanie odwrotnego zadania kinematyki uzyskujemy przez scałkowanie układu równań różniczkowych

$$\dot{\mathbf{x}}_d = (\mathbf{J}^{ae})^{-1}(\mathbf{x}_d) \begin{pmatrix} \dot{\mathbf{y}}_d \\ \mathbf{0} \end{pmatrix}, \quad (3.40)$$

przy warunku początkowym  $\mathbf{x}_0 = \lim_{t \rightarrow +\infty} \xi(t)$ , gdzie  $\xi(t)$  jest trajektorią układu

$$\dot{\xi} = -\alpha(\mathbf{J}^{ae})^{-1}(\xi) \begin{pmatrix} \mathbf{k}(\xi) - \mathbf{y}_0 \\ \mathbf{0} \end{pmatrix}. \quad (3.41)$$

Połączenie wyrażeń (3.40) i (3.41) daje *asymptotyczny algorytm kinematyki odwrotnej*

$$\dot{\mathbf{x}} = (\mathbf{J}^{ae})^{-1}(\mathbf{x}) \begin{pmatrix} \dot{\mathbf{y}}_d - \alpha(\mathbf{k}(\mathbf{x}) - \mathbf{y}_d) \\ \mathbf{0} \end{pmatrix}, \quad (3.42)$$

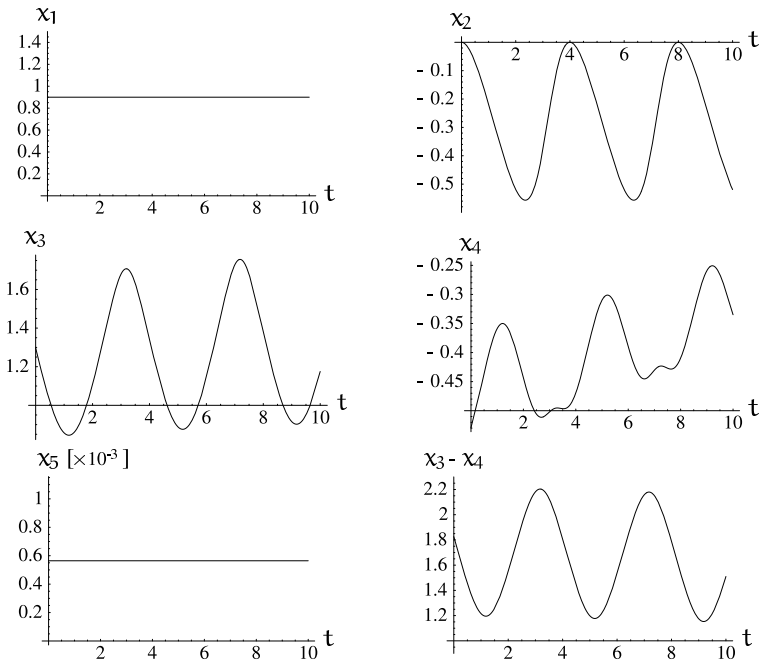
zbieżny do rozwiązania zadania odwrotnego przy  $t \rightarrow +\infty$ .

### Przykład 3.1.10 (Manipulator IRb-6 na torze jezdny)

Przyjmijmy ponownie, że zadanie manipulatora IRb-6 zamontowanego na torze jezdny polega na wykonaniu ruchu efektora wzdłuż trajektorii  $\mathbf{y}_d(t)$  zadanej wzorem (3.24) dla chwil czasu  $t \in \mathcal{I} = [0, 10][s]$ . Dodatkowo zażądajmy, aby ruch ten został zrealizowany bez poruszania pierwszym, piątym i szóstym przegubem manipulatora, co można zapisać w formie ograniczeń konfiguracyjnych (3.37) jako

$$x_1 = x_{10}, \quad x_5 = x_{50}, \quad x_6 = x_{60},$$

gdzie  $x_{i0}$  oznacza  $i$ -tą składową konfiguracji początkowej manipulatora. Dołączając powyższe równania do równań kinematyki manipulatora (2.77) i różniczkując całość otrzymujemy jacobian rozszerzony manipulatora  $\mathbf{J}^{ae}(\mathbf{x})$  odpowiadający wybranym ograniczeniom. Rozwiązanie układu równań różniczkowych (3.40), z tak wyliczonym jacobianem, zainicjowanego w punkcie początkowym  $\mathbf{x}(0) = (0.9, 0., 1.29, -0.529, 0., 0.)^T$ , przedstawiono na rysunku 3.18 (nie pokazany kąt  $x_6$  był stale równy 0). Jak widać z rysunku, otrzymana trajektoria spełnia nałożone ograniczenia. ■



**Rysunek 3.18** Rozwiązanie zadania odwrotnego dla manipulatora IRb-6 zamontowanego na torze jezdnyim uzyskane przy wykorzystaniu algorytmu jacobianu rozszerzonego.

### Metoda jacobianu transponowanego

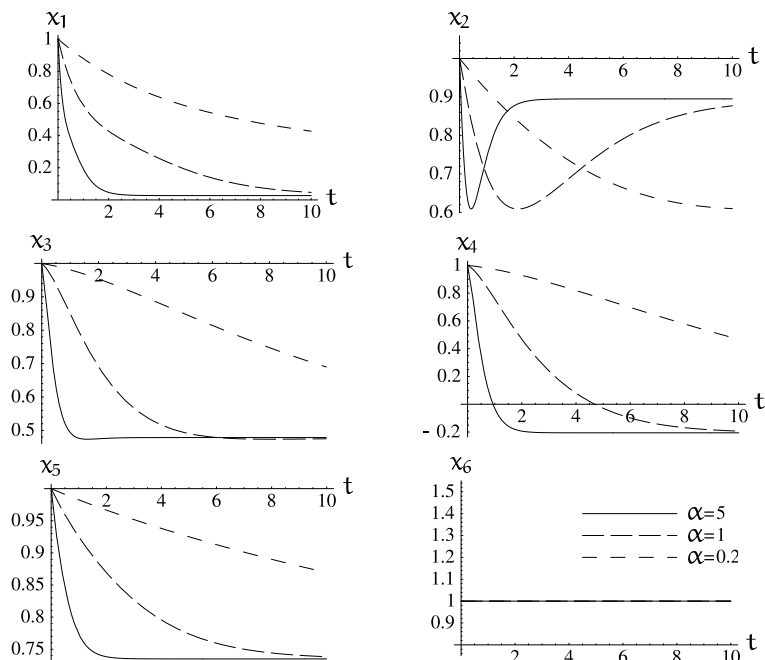
Niech teraz będzie dane punktowe zadanie odwrotne, polegające na obliczeniu konfiguracji przegubów  $\mathbf{x}_d$  odpowiadającej umieszczeniu efektora w określonym punkcie  $\mathbf{y}_d$  przestrzeni zadaniowej. Zauważmy, że do osiągnięcia tego celu dobrze nadają się algorytmy (3.21), (3.29) i (3.41) zainicjowane przy dowolnym  $\xi_0 \in \mathbb{R}^n$ . Co więcej, przy założeniu, że jacobian analityczny jest pełnego rzędu, tzn.  $\text{rank } \mathbf{J}^a(\mathbf{x}) = m$ , algorytm

$$\dot{\mathbf{x}} = -\alpha \mathbf{J}^{aT}(\mathbf{x})(\mathbf{k}(\mathbf{x}) - \mathbf{y}_d), \quad (3.43)$$

przy  $\alpha > 0$ , jest algorytmem *najszybszego spadku* zapewniającym ruch w przestrzeni przegubowej w kierunku przeciwnym do gradientu funkcji kwadratu błędu

$$\mathbf{V}(\mathbf{x}) = \frac{1}{2}(\mathbf{k}(\mathbf{x}) - \mathbf{y}_d)^T(\mathbf{k}(\mathbf{x}) - \mathbf{y}_d).$$

Algorytm (3.43) jest algorytmem *metody jacobianu transponowanego*. Trajektoria układu (3.43) zmierza w kierunku punktów równowagi speł-



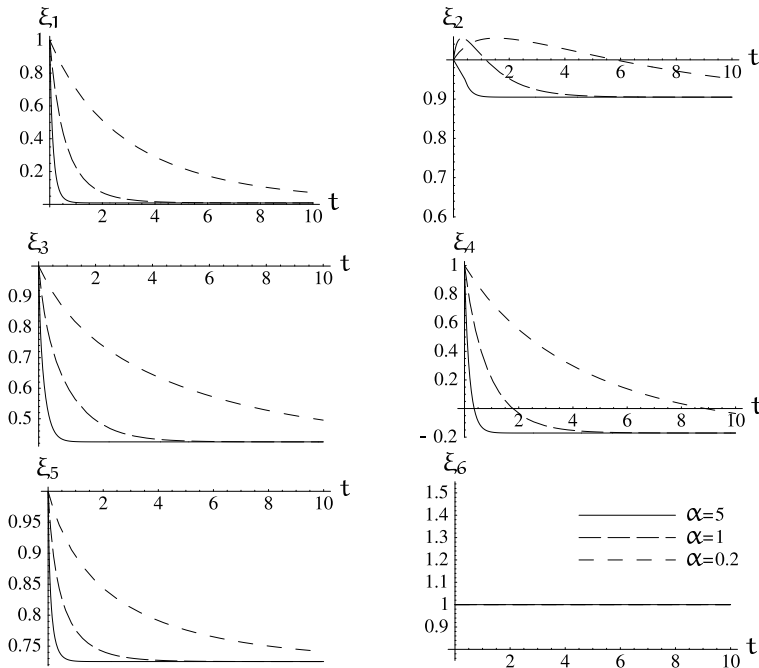
**Rysunek 3.19** Rozwiązanie punktowego zadania odwrotnego dla manipulatora IRb-6 zamontowanego na torze jezdnyim uzyskane przy wykorzystaniu algorytmu jakobianu transponowanego.

niających zależność  $\mathbf{J}^{\text{aT}}(\mathbf{x})(\mathbf{k}(\mathbf{x}) - \mathbf{y}_d) = \mathbf{0}$ . Łatwo zauważyć, że o ile tylko  $\text{rank } \mathbf{J}^{\text{a}}(\mathbf{x}) = m$ , punktami  $\mathbf{x}$  równowagi układu (3.43) są rozwiązania zadania odwrotnego  $\mathbf{k}(\mathbf{x}) = \mathbf{y}_d$ . Jest rzeczą oczywistą, że z obliczeniowego punktu widzenia algorytm najszybszego spadku (3.43) jest znacznie mniej złożony niż algorytm (3.29).

### Przykład 3.1.11 (Manipulator IRb-6 na torze jezdnyim)

W celu zilustrowania działania algorytmu (3.43) założmy, że naszym zadaniem jest znalezienie punktu startowego  $\mathbf{x}_0$  dla trajektorii (3.24) (tzn. takiego  $\mathbf{x}_0$ , że  $\mathbf{k}(\mathbf{x}_0) = \mathbf{y}(0)$  przy kinematyce  $\mathbf{k}(\mathbf{x})$  opisanej równaniem (2.77)). W tym celu rozwiązaliśmy układ równań różniczkowych (3.43) zainicjowany w punkcie  $\mathbf{x}(0) = (1, 1, 1, 1, 1, 1)^{\text{T}}$ . Wyniki, przy różnych wartościach parametru  $\alpha$ , przedstawia rysunek 3.19.

Analogicznych obliczeń dokonaliśmy posługując się algorytmem (3.29) wykorzystującym pseudoodwrotność jakobianu analitycznego manipulatora (macierz  $\mathbf{W} = \mathbb{I}_6$ ). Rezultat jest pokazany na rysunku 3.20. Po porównaniu



**Rysunek 3.20** Rozwiązanie punktowego zadania odwrotnego dla manipulatora IRb-6 zamontowanego na torze jezdny uzyskane przy wykorzystaniu pseudoodwrotności jacobianu.

obu rysunków łatwo zauważyć, że algorytm z pseudoodwrotnością jacobianu charakteryzuje się szybszą zbieżnością. Ceną, jaką trzeba za to zapłacić, jest czas obliczeń. Czas znajdowania rozwiązania z użyciem jacobianu transponowanego był prawie o trzy rzędy krótszy od czasu potrzebnego na uzyskanie rozwiązania przy pomocy algorytmu (3.29) przy takich samych wartościach parametrów. ■

## Geometryczne metody jacobianowe

Algorytmy kinematyki odwrotnej we współrzędnych, przedstawione formułami (3.20), (3.28), (3.33) i (3.40), można zaadaptować do rozwiązania zadania odwrotnego postawionego w sposób ogólny, bez odwołania do układów współrzędnych. W tym celu założymy, że jest dana trajektoria  $\mathbf{z}_d(t) \in \mathcal{Z} \subset \text{SE}(3)$  na rozmaitości zadaniowej, określona w chwilach  $t \in \mathcal{I} \subset \mathbb{R}$ . Aby znaleźć odpowiednią trajektorię ruchu przegubów  $\mathbf{q}_d(t)$ , wyliczamy



prędkość w układzie przestrzeni

$$\mathbf{v}_d^s = \begin{bmatrix} [\boldsymbol{\omega}_{sd}] & \mathbf{v}_{sd} \\ \mathbf{0} & 0 \end{bmatrix} = \dot{\mathbf{z}}_d(t) \mathbf{z}_d^{-1}(t)$$

i wyznaczamy wektory  $\boldsymbol{\omega}_{sd}$ ,  $\mathbf{v}_{sd}$ . Związek między prędkością  $\dot{\mathbf{q}}$  ruchu przegubów a wektorem prędkości  $(\mathbf{v}_s^T, \boldsymbol{\omega}_s^T)^T$  jest określony przez jakobian geometryczny w przestrzeni

$$\begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\omega}_s \end{pmatrix} = \mathbf{J}^s(\mathbf{q}) \dot{\mathbf{q}}. \quad (3.44)$$

Odpowiednikiem algorytmu (3.33) jest w tym przypadku algorytm

$$\dot{\mathbf{q}}_d = \mathbf{J}_W^{s\#}(\mathbf{q}_d) \begin{pmatrix} \mathbf{v}_{sd} \\ \boldsymbol{\omega}_{sd} \end{pmatrix} + (\mathbb{I}_n - \mathbf{J}_W^{s\#}(\mathbf{q}_d) \mathbf{J}^s(\mathbf{q}_d)) \mathbf{v}. \quad (3.45)$$

Analogicznie, dodając do kinematyki  $\mathbf{K}(\mathbf{q})$  manipulatora  $n - m$  niezależnych ograniczeń, jak w formule (3.37), otrzymujemy rozszerzony jakobian geometryczny w przestrzeni

$$\mathbf{J}^{se}(\mathbf{q}) = \begin{bmatrix} \mathbf{J}^s(\mathbf{q}) \\ \frac{\partial \mathbf{f}}{\partial \mathbf{q}}(\mathbf{q}) \end{bmatrix} \quad (3.46)$$

i formułujemy odpowiednik algorytmu (3.40) w postaci

$$\dot{\mathbf{q}}_d = (\mathbf{J}^{se})^{-1}(\mathbf{q}_d) \begin{pmatrix} \mathbf{v}_{sd} \\ \boldsymbol{\omega}_{sd} \\ \mathbf{0} \end{pmatrix}. \quad (3.47)$$

### 3.1.4 Metoda mnożników Lagrange'a

Rozważmy kinematykę manipulatora redundantnego wyrażoną we współrzędnych

$$\mathbf{y} = \mathbf{k}(\mathbf{x}),$$

gdzie  $\mathbf{x} \in \mathbb{R}^n$ ,  $\mathbf{y} \in \mathbb{R}^m$ ,  $m < n$ , i założmy, że punktowe zadanie odwrotne polega na znalezieniu konfiguracji przegubów odpowiadającej osiągnięciu przez

efektor ustalonych współrzędnych zadaniowych  $\mathbf{y}_d$ . Metoda mnożników Lagrange'a polega na wprowadzeniu pewnej funkcji oceny jakości zachowania manipulatora  $H(\mathbf{x})$ , określonej na przestrzeni przegubowej, i sprowadzeniu zadania odwrotnego do zadania optymalizacji warunkowej funkcji  $H(\mathbf{x})$  przy ograniczeniu równościowym

$$\mathbf{F}(\mathbf{x}) = \mathbf{k}(\mathbf{x}) - \mathbf{y}_d = \mathbf{0}. \quad (3.48)$$

Standardowa metoda rozwiązywania takiego zadania wymaga zdefiniowania lagranżianu

$$L(\mathbf{x}, \boldsymbol{\lambda}) = H(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{F}(\mathbf{x})$$

zawierającego wektor mnożników Lagrange'a  $\boldsymbol{\lambda} \in \mathbb{R}^m$ . Warunki konieczne optymalności mają postać

$$\mathbf{J}^{\alpha T}(\mathbf{x})\boldsymbol{\lambda} = \mathbf{h}(\mathbf{x}), \quad (3.49)$$

przy oznaczeniach  $\mathbf{J}^{\alpha}(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x})$ ,  $\mathbf{h}(\mathbf{x}) = -\frac{\partial H}{\partial \mathbf{x}}(\mathbf{x})$ . Załóżmy, że pierwsze  $m$  kolumn jacobianu analitycznego  $\mathbf{J}^{\alpha}(\mathbf{x})$  jest niezależnych. W celu wyeliminowania z układu równań (3.49) wektora  $\boldsymbol{\lambda}$  podzielimy te równania na dwie grupy złożone, odpowiednio, z  $m$  pierwszych równań oraz z pozostałych  $n - m$  równań, w następujący sposób

$$\mathbf{J}^{\alpha T}(\mathbf{x})\boldsymbol{\lambda} = \begin{bmatrix} \mathbf{J}_m(\mathbf{x}) \\ \mathbf{J}_{n-m}(\mathbf{x}) \end{bmatrix} \boldsymbol{\lambda} = \begin{pmatrix} \mathbf{h}_m(\mathbf{x}) \\ \mathbf{h}_{n-m}(\mathbf{x}) \end{pmatrix}.$$

Do wyliczenia mnożników Lagrange'a wykorzystamy pierwszą grupę równań otrzymując

$$\boldsymbol{\lambda} = \mathbf{J}_m^{-1}(\mathbf{x})\mathbf{h}_m(\mathbf{x}). \quad (3.50)$$

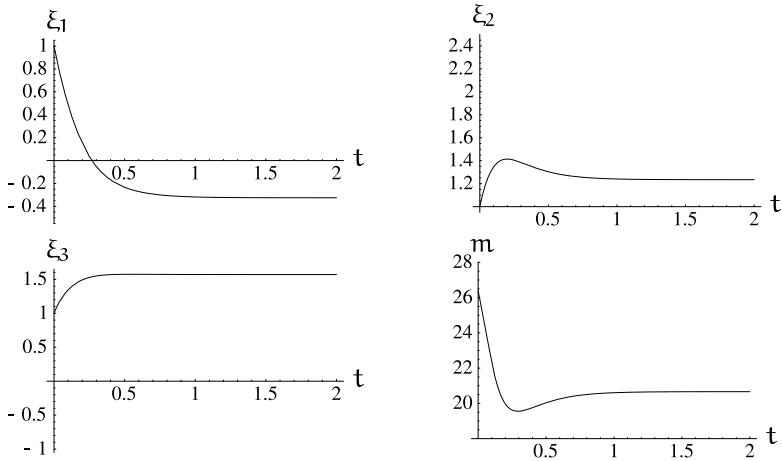
Podstawienie (3.50) do drugiej grupy równań pozwala uzyskać następującą zależność

$$\begin{bmatrix} -\mathbf{J}_{n-m}(\mathbf{x})\mathbf{J}_m^{-1}(\mathbf{x}) & \mathbb{I}_{n-m} \end{bmatrix} \mathbf{h}(\mathbf{x}) = \mathbf{0}. \quad (3.51)$$

Rozwiązanie odwrotnego zadania kinematyki jest teraz równoważne rozwiązaniu układu  $n$  równań złożonego z (3.48) oraz (3.51).

### Przykład 3.1.12 (Potrójne wahadło)

Szukamy rozwiązania  $\mathbf{x}_0$  punktowego zadania odwrotnego dla potrójnego wahadła z przykładu 3.1.7, dla punktu  $\mathbf{y}_{d0} = (5, 2)^T$ , który jest punktem

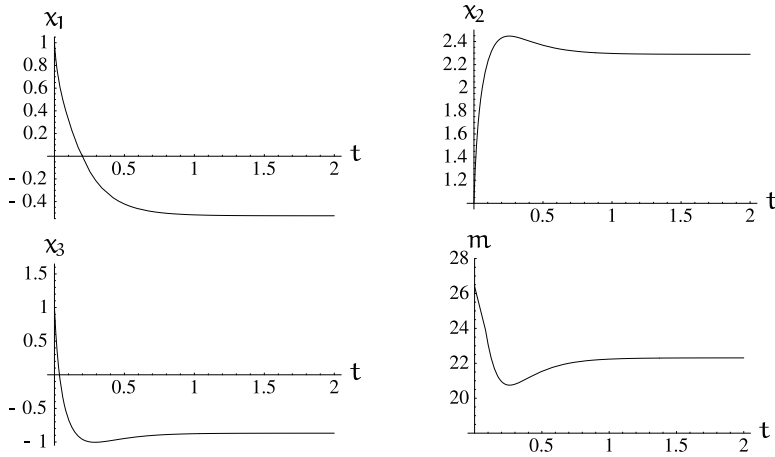


**Rysunek 3.21** Rozwiązanie punktowego zadania odwrotnego dla potrójnego wadła uzyskane przy wykorzystaniu pseudoodwrotności jacobianu.

startowym trajektorii (3.31). Zadanie to można rozwiązać tak samo jak w przykładzie 3.1.7, wykorzystując układ równań różniczkowych (3.29). Przebieg trajektorii układu (3.29) dla  $\alpha = 5$ ,  $\mathbf{W} = \mathbb{I}_3$  oraz warunku początkowego  $\boldsymbol{\xi} = (1, 1, 1)^\top$  przedstawia rysunek 3.21. Na rysunku pokazaliśmy także przebieg funkcji manipulowalności manipulatora  $m(\mathbf{x}(t))^*$  wzdłuż otrzymanej trajektorii. Jako rozwiązanie zadania punktowego otrzymujemy w ten sposób konfigurację  $\mathbf{x}_0 = (-0.325, 1.23, 1.57)^\top$ , dla której manipulowalność wynosi  $m(\mathbf{x}_0) = 20.7$ .

Podany wyżej sposób pozwala na znalezienie jednego z wielu rozwiązań zadania odwrotnego. Jeżeli interesuje nas rozwiązanie spełniające dodatkowe kryteria, do jego określenia możemy posłużyć się metodą mnożników Lagrange'a z odpowiednio dobraną funkcją oceny jakości zachowania manipulatora  $H(\mathbf{x})$ . Załóżmy, że zależy nam na znalezieniu rozwiązania położonego możliwie daleko od konfiguracji osobliwych. W tym celu funkcję  $H(\mathbf{x})$  przyjmiemy w postaci kwadratu funkcji manipulowalności  $m(\mathbf{x})$ . Po dokonaniu takiego wyboru, w wyniku rozwiązania układu równań (3.48), (3.51) za pomocą algorytmu Newtona otrzymujemy przebiegi położenia przegubów manipulatora i manipulowalności przedstawione na rysunku 3.22. Rozwiązaniem zadania punktowego jest wektor położenia przegubów manipulatora  $\mathbf{x}_0 = (-0.525, 2.29, -0.869)^\top$ , dla którego manipulowalność osiąga wartość  $m(\mathbf{x}_0) = 22.3$ . ■

\*Zdefiniowanej w następnym podrozdziale.



Rysunek 3.22 Rozwiązanie punktowego zadania odwrotnego dla potrójnego waha dla uzyskane metodą mnożników Lagrange'a.

### 3.1.5 Elipsoida manipulowalności

Z definicji jacobianu analitycznego lub geometrycznego manipulatora wynika, że obydwa typy jacobianów można traktować jako zależne od bieżącej konfiguracji przekształcenia liniowe prędkości ruchu w przegubach w prędkość ruchu efektora. Dla ustalenia uwagi rozważmy jacobian analityczny  $\mathbf{J}^a(\mathbf{x})$  jako przekształcenie prędkości

$$\dot{\mathbf{y}} = \mathbf{J}^a(\mathbf{x})\dot{\mathbf{x}}.$$

Zdefiniujmy sferę jednostkową w przestrzeni prędkości przegubowych

$$S = \{\dot{\mathbf{x}} \mid \|\dot{\mathbf{x}}\|^2 = 1\}$$

i wyznaczmy jej obraz w przestrzeni prędkości zadaniowych korzystając z zależności (3.28) przy założeniu, że macierz  $\mathbf{W} = \mathbb{I}_n$ . W wyniku otrzymujemy elipsoidę

$$\mathbf{E}_m(\mathbf{x}) = \left\{ \dot{\mathbf{y}} \mid \dot{\mathbf{y}}^T (\mathbf{J}^a(\mathbf{x})\mathbf{J}^{aT}(\mathbf{x}))^{-1} \dot{\mathbf{y}} = 1 \right\}, \quad (3.52)$$

zwaną *elipsoidą manipulowalności*. Macierz

$$\mathbf{M}(\mathbf{x}) = \mathbf{J}^a(\mathbf{x})\mathbf{J}^{aT}(\mathbf{x})$$

rozmiaru  $m \times m$  nazywamy *macierzą manipulowalności*. Z definicji ma-

cierzy manipulowalności i twierdzenia Rayleigha-Ritza\* wynika następujące oszacowanie

$$\underline{\lambda}_{\mathbf{M}(\mathbf{x})} \|\mathbf{v}\|^2 \leq \mathbf{v}^T \mathbf{M}(\mathbf{x}) \mathbf{v} \leq \bar{\lambda}_{\mathbf{M}(\mathbf{x})} \|\mathbf{v}\|^2,$$

gdzie  $\underline{\lambda}_{\mathbf{M}(\mathbf{x})}$ ,  $\bar{\lambda}_{\mathbf{M}(\mathbf{x})}$  oznaczają, odpowiednio, najmniejszą i największą wartość własną macierzy  $\mathbf{M}(\mathbf{x})$ . W konsekwencji, elipsoida (3.52) zawiera się w pierścieniu sferycznym ograniczonym dwiema sferami

$$\|\dot{\mathbf{y}}\|^2 = \underline{\lambda}_{\mathbf{M}(\mathbf{x})} \quad \text{oraz} \quad \|\dot{\mathbf{y}}\|^2 = \bar{\lambda}_{\mathbf{M}(\mathbf{x})}$$

o promieniach  $\sqrt{\underline{\lambda}_{\mathbf{M}(\mathbf{x})}}$ ,  $\sqrt{\bar{\lambda}_{\mathbf{M}(\mathbf{x})}}$ . Elipsoida manipulowalności obrazuje zdolności manipulacyjne manipulatora zwane zręcznością manipulatora; w szczególności to, na ile elipsoida odbiega od sfery wskazuje na *stopień anizotropii* działania manipulatora w danej konfiguracji. W konfiguracji izotropowej, tzn. gdy  $\underline{\lambda}_{\mathbf{M}(\mathbf{x})} = \bar{\lambda}_{\mathbf{M}(\mathbf{x})}$ , elipsoida manipulowalności staje się sferą. Miarą liczbową manipulowalności w danej konfiguracji jest objętość elipsoidy manipulowalności<sup>†</sup>

$$m(\mathbf{x}) = \sqrt{\det \mathbf{M}(\mathbf{x})}, \quad (3.53)$$

która w przypadku manipulatora nieredundantnego jest równa

$$m(\mathbf{x}) = |\det \mathbf{J}^a(\mathbf{x})|. \quad (3.54)$$

Funkcja  $m(\mathbf{x})$  nazywa się *funkcją manipulowalności* manipulatora. Jak wynika z definicji (3.53), manipulowalność w konfiguracji  $\mathbf{x}$  jest równa pierwiastkowi kwadratowemu z iloczynu wartości własnych macierzy manipulowalności,

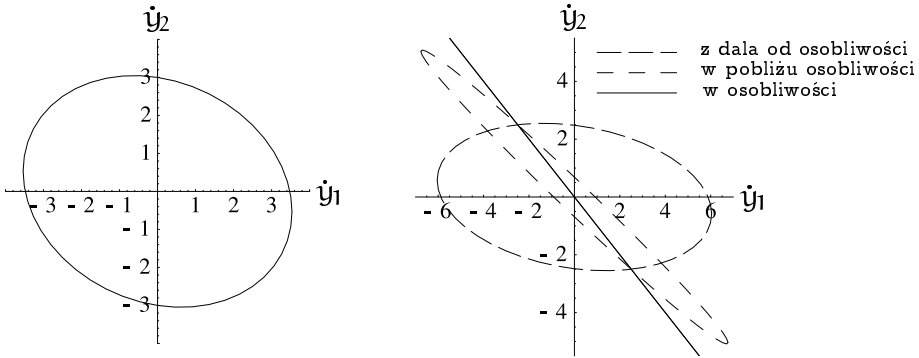
$$m(\mathbf{x}) = \sqrt{\prod_{i=1}^m \lambda_{\mathbf{M}(\mathbf{x})i}}. \quad (3.55)$$

Zgodnie z oczekiwaniem, manipulowalność maleje do zera w konfiguracjach osobliwych.

---

\*Zobacz dodatek A.1.

<sup>†</sup>Ściśle, objętość elipsoidy manipulowalności =  $m(\mathbf{x})V(\mathbb{S}^{m-1})$ , gdzie  $V(\mathbb{S}^{m-1})$  oznacza objętość sfery jednostkowej w  $\mathbb{R}^m$ .



**Rysunek 3.23** Przykładowa elipsoida manipulowalności dla manipulatora typu podwójne wahadło oraz wpływ odległości od konfiguracji osobliwej na kształt elipsoidy.

Wartości własne macierzy manipulowalności mogą posłużyć do zdefiniowania innych miar zręczności manipulatora. Do ważniejszych miar należy tzw. *stopień uwarunkowania* konfiguracji  $\mathbf{x}$ ,

$$\kappa(\mathbf{x}) = \sqrt{\frac{\bar{\lambda}_{\mathbf{M}(\mathbf{x})}}{\underline{\lambda}_{\mathbf{M}(\mathbf{x})}}}, \quad (3.56)$$

którego duże wartości wskazują na znaczny stopień anizotropii konfiguracji  $\mathbf{x}$ . Dla konfiguracji bliskich osobliwym stopień uwarunkowania rośnie nieograniczenie. Funkcje  $m(\mathbf{x})$ ,  $\kappa(\mathbf{x})$  mogą być wykorzystywane w algorytmach kinematyki odwrotnej typu (3.33) do uzyskania trajektorii  $\mathbf{x}(t)$  zapewniającej dużą manipulowalność lub izotropowość manipulatora. W tym celu za wektor  $\mathbf{v}$  w wyrażeniu (3.33) należy podstawić odpowiednio

$$\mathbf{v} = \beta \frac{\partial m(\mathbf{x})}{\partial \mathbf{x}} \quad \text{lub} \quad \mathbf{v} = -\beta \frac{\partial \kappa(\mathbf{x})}{\partial \mathbf{x}}, \quad \beta > 0.$$

### Przykład 3.1.13 (Manipulator typu podwójne wahadło)

Rysunek 3.23 przedstawia przykładową elipsoidę<sup>‡</sup> manipulowalności dla manipulatora typu podwójne wahadło (wykres po lewej stronie), oraz wpływ odległości od konfiguracji osobliwej na kształt tej elipsoidy (wykres po prawej stronie). Jak widać, w miarę zbliżania się do osobliwej, powierzchnia ograniczona elipsą maleje, by w konfiguracji osobliwej spaść do zera (elipsa degeneruje się do odcinka). ■

<sup>‡</sup>W istocie — elipsę.

## 3.2 Osobliwe odwrotne zadanie kinematyki

We wszystkich rozważanych dotąd algorytmach kinematyki odwrotnej wymagana była nieosobliwość jacobianu manipulatora, czyli pełny rząd wierszowy macierzy  $\mathbf{J}^a(\mathbf{x})$ ,  $\mathbf{J}^s(\mathbf{q})$ ,  $\mathbf{J}^b(\mathbf{q})$ ,  $\mathbf{J}^m(\mathbf{q})$ . Występowanie konfiguracji osobliwych manipulatora ma wpływ na istnienie i jednoznaczność rozwiązania odwrotnego zadania kinematyki. Ze względu na złe uwarunkowanie operacji wyznaczania odwrotności (pseudoodwrotności) jacobianu manipulatora w pobliżu konfiguracji osobliwych, co może prowadzić do nieograniczonego wzrostu prędkości ruchu przegubów manipulatora, osobliwe zadanie odwrotne kinematyki wymaga zastosowania specjalnych algorytmów.

### 3.2.1 Unikanie osobliwości

#### Pojęcia podstawowe

Istnienie konfiguracji osobliwych manipulatorów posiadających nieograniczone przeguby obrotowe ma źródło w odmiennej geometrii rozmaitości konfiguracyjnej i rozmaitości zadaniowej manipulatora. Ta przyczyna osobliwości kinematyki ma charakter fundamentalny i nie może być usunięta poprzez przekonstruowanie manipulatora (z wyjątkiem mechanicznego ograniczenia zakresu ruchu w przegubach obrotowych). W szczególności, zwiększenie liczby stopni swobody manipulatora (wprowadzenie redundancji) nie eliminuje konfiguracji osobliwych, chociaż powoduje, że ich obecność staje się mniej uciążliwa. Okazuje się bowiem, że redundancja manipulatora może zostać wykorzystana do uniknięcia pewnych konfiguracji osobliwych przy realizacji zadania manipulatora. Aby przedstawić zagadnienie unikania konfiguracji osobliwych w sposób precyzyjny, załóżmy, że jest dana reprezentacja kinematyki manipulatora redundantnego ( $n > m$ ) we współrzędnych (2.65),

$$\mathbf{k} : \mathbb{R}^n \longrightarrow \mathbb{R}^m, \quad \mathbf{y} = \mathbf{k}(\mathbf{x}). \quad (3.57)$$

Konfiguracje osobliwe kinematyki, traktowanej jako odwzorowanie, należą do zbioru

$$S = \left\{ \mathbf{x} \in \mathbb{R}^n \mid \text{rank} \left( \mathbf{J}^a(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}) \right) < m \right\}.$$

Powiadamy, że konfiguracja osobliwa  $\mathbf{x} \in S$  jest *możliwa do uniknięcia*, jeżeli istnieje nieosobliwa konfiguracja  $\mathbf{x}'$  manipulatora, taka że  $\mathbf{k}(\mathbf{x}) = \mathbf{k}(\mathbf{x}')$ .

W przeciwnym razie, konfiguracja  $\mathbf{x}$  jest *niemożliwa do uniknięcia*. Własność „możliwa do uniknięcia” oznacza, że położenie i orientacja efektora w przestrzeni zadaniowej, osiągalne w konfiguracji osobliwej, mogą być zrealizowane w innej, nieosobliwej konfiguracji manipulatora. Jeżeli ta konfiguracja nieosobliwa leży w pewnym (małym) otoczeniu konfiguracji osobliwej  $\mathbf{x}$ , mówimy że konfiguracja  $\mathbf{x}$  jest *lokalnie możliwa do uniknięcia*. Jak łatwo zauważyć, własność „lokalnie możliwa do uniknięcia” jest mocniejsza od własności „możliwa do uniknięcia”. Jeżeli w pewnym otoczeniu konfiguracji  $\mathbf{x}$  nie ma nieosobliwej konfiguracji  $\mathbf{x}'$ , takiej że  $\mathbf{k}(\mathbf{x}) = \mathbf{k}(\mathbf{x}')$ , to  $\mathbf{x}$  nazywamy konfiguracją *lokalnie niemożliwą do uniknięcia*. Ta ostatnia własność jest w oczywisty sposób słabsza od własności „niemożliwa do uniknięcia”.

### Kinematyka o stopniu redundancji 1

Kryteria własności „konfiguracja osobliwa możliwa do uniknięcia” przedstawimy najpierw przy założeniu, że stopień redundancji kinematyki (3.57) wynosi 1, tzn. gdy  $n = m + 1$ . W tym celu stowarzyszymy z kinematyką pewien układ dynamiczny

$$\dot{\mathbf{x}} = \mathbf{X}(\mathbf{x}), \quad (3.58)$$

zwany *układem dynamicznym ruchu własnego*, w którym poszczególne składowe pola wektorowego  $\mathbf{X}(\mathbf{x}) = (X_1(\mathbf{x}), \dots, X_n(\mathbf{x}))^T$  definiuje się formułą

$$X_i(\mathbf{x}) = (-1)^{i+1} \det \mathbf{J}^{ai}(\mathbf{x}), \quad (3.59)$$

a symbol  $\mathbf{J}^{ai}(\mathbf{x})$  oznacza jakobian analityczny  $\mathbf{J}^a(\mathbf{x})$  z usuniętą  $i$ -tą kolumną.

Można pokazać, że układ dynamiczny ruchu własnego został zdefiniowany w taki sposób, że  $\mathbf{X}(\mathbf{x}) \in \text{Ker } \mathbf{J}^a(\mathbf{x})$ , pole  $\mathbf{X}$  ma zerową dywergencję,

$$\text{div } \mathbf{X}(\mathbf{x}) = \text{tr } \frac{\partial \mathbf{X}}{\partial \mathbf{x}}(\mathbf{x}) = 0,$$

a jego punkty równowagi  $\mathbf{X}(\mathbf{x}) = \mathbf{0}$  znajdują się w konfiguracjach osobliwych. Wynika stąd, że trajektoria układu (3.58), różna od punktu równowagi, składa się z konfiguracji nieosobliwych. Co więcej, wzdłuż trajektorii tego układu kinematyka  $\mathbf{k}(\mathbf{x})$  przyjmuje stałą wartość. Oznacza to, że trajektorie leżą na tzw. *rozmaitości ruchu własnego* kinematyki. Jeżeli pewna trajektoria zbliża się asymptotycznie przy  $t \rightarrow \pm\infty$  do punktu równowagi układu



dynamicznego ruchu własnego, oznacza to, że wartości współrzędnych zadaniowych przyjmowane przez kinematykę manipulatora w punkcie równowagi i na trajektorii są takie same. W konsekwencji, konfiguracja osobliwa odpowiadająca punktowi równowagi układu dynamicznego ruchu własnego jest możliwa do uniknięcia. Jeżeli natomiast punkt równowagi nie posiada dążących do/od niego trajektorii układu (3.58), to odpowiadająca mu konfiguracja osobliwa jest niemożliwa do uniknięcia. Wyrażając to spostrzeżenie w sposób formalny powiemy, że konfiguracja osobliwa  $\mathbf{x}$  jest możliwa do uniknięcia, jeżeli punkt  $\mathbf{x}$  jest asymptotycznie stabilnym lub niestabilnym punktem równowagi układu (3.58). Konfiguracja  $\mathbf{x}$  jest niemożliwa do uniknięcia, jeżeli  $\mathbf{x}$  jest punktem równowagi układu (3.58), stabilnym w sensie Lapunowa. Dzięki takiemu sformułowaniu, rozstrzygnięcie czy dana konfiguracja osobliwa manipulatora jest możliwa czy niemożliwa do uniknięcia zostało sprowadzone do badania stabilności układu dynamicznego ruchu własnego. Elementarnym warunkiem wystarczającym na to, aby konfiguracja osobliwa  $\mathbf{x}$  była lokalnie możliwa do uniknięcia (a więc możliwa do uniknięcia) jest, by macierz przybliżenia liniowego układu (3.58) w punkcie równowagi posiadała wartość własną o różnej od zera części rzeczywistej.

Badanie stabilności układu (3.58) upraszcza się znacznie przy odpowiednim wyborze współrzędnych przegubowych. Niech  $\mathbf{x}_0 = (x_{01}, \dots, x_{0m-1}, x_{0m}, x_{0n})^T$  oznacza konfigurację osobliwą kinematyki (3.57) korzędu 1, co oznacza, że jacobian analityczny  $\mathbf{J}^a(\mathbf{x}_0) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}_0)$  ma rząd  $m-1$ . Bez utraty ogólności możemy założyć, że w konfiguracji  $\mathbf{x}_0$  jacobian analityczny spełnia następujący warunek

$$\text{rank} \begin{bmatrix} \frac{\partial k_1}{\partial x_1} & \dots & \frac{\partial k_1}{\partial x_{m-1}} \\ \vdots & \ddots & \vdots \\ \frac{\partial k_{m-1}}{\partial x_1} & \dots & \frac{\partial k_{m-1}}{\partial x_{m-1}} \end{bmatrix}(\mathbf{x}_0) = m-1. \quad (3.60)$$

Korzystając z własności (3.60) zdefiniujemy w pewnym otoczeniu  $\mathbf{x}_0$  nowy układ współrzędnych

$$\boldsymbol{\xi} = \boldsymbol{\varphi}(\mathbf{x}) = (k_1(\mathbf{x}), \dots, k_{m-1}(\mathbf{x}), x_m, x_n)^T. \quad (3.61)$$

Nietrudno sprawdzić, że  $\text{rank} \frac{\partial \boldsymbol{\varphi}}{\partial \mathbf{x}}(\mathbf{x}_0) = n$ , zatem, na mocy twierdzenia o funkcji odwrotnej\*, (3.61) jest odwzorowaniem lokalnie wzajemnie jednoznaczny i gładkim (tzn. lokalnym dyfeomorfizmem). W nowych współ-

\*Zobacz dodatek A.2.

rzędnych kinematyka  $\mathbf{y} = \mathbf{k}(\mathbf{x})$  przyjmuje postać

$$\mathbf{y} = \mathbf{k} \circ \boldsymbol{\varphi}^{-1}(\boldsymbol{\xi}) = \mathbf{h}(\boldsymbol{\xi}) = (\xi_1, \dots, \xi_{m-1}, h_m(\boldsymbol{\xi}))^T,$$

w której funkcja  $h_m$  spełnia warunek  $h_m \circ \boldsymbol{\varphi}(\mathbf{x}) = k_m(\mathbf{x})$ . Układ dynamiczny ruchu własnego upraszcza się w nowych współrzędnych do postaci ( $n = m + 1$ )

$$\begin{cases} \dot{\xi}_1 = 0 & \dots & \dot{\xi}_{m-1} = 0 \\ \dot{\xi}_m = (-1)^{m+1} \frac{\partial h_m(\boldsymbol{\xi})}{\partial \xi_n} \\ \dot{\xi}_n = (-1)^{m+2} \frac{\partial h_m(\boldsymbol{\xi})}{\partial \xi_m}. \end{cases} \quad (3.62)$$

Co więcej, ponieważ współrzędne  $\xi_1 = k_1(\mathbf{x}_0), \dots, \xi_{m-1} = k_{m-1}(\mathbf{x}_0)$  są stałe na trajektoriach układu, dynamika układu (3.62) redukuje się do dynamiki układu hamiltonowskiego w  $\mathbb{R}^2$

$$\dot{q} = \frac{\partial H(q, p)}{\partial p}, \quad \dot{p} = -\frac{\partial H(q, p)}{\partial q}, \quad (3.63)$$

z hamiltonianem  $H(q, p) = (-1)^{m+1} h_m(k_1(\mathbf{x}_0), \dots, k_{m-1}(\mathbf{x}_0), q, p)^\dagger$ .

Punktem równowagi układu (3.63) jest  $q_0 = x_{0m}, p_0 = x_{0n}$ . Ze względu na to, że punkty równowagi układu hamiltonowskiego są punktami krytycznymi hamiltonianu  $H(q, p)$ , stabilne w sensie Lapunowa będą te punkty, w których hamiltonian osiąga minimum bądź maksimum, natomiast niestabilne są punkty, w których hamiltonian ma punkt siodłowy. Wynika stąd następujące kryterium.

**Twierdzenie 3.2.1** *Niech  $\mathbf{x}_0 = (x_{01}, \dots, x_{0m}, x_{0n})^T \in \mathbb{R}^n$  oznacza konfigurację osobliwą korzędu 1 kinematyki  $\mathbf{k}(\mathbf{x}) = (k_1(\mathbf{x}), k_2(\mathbf{x}), \dots, k_m(\mathbf{x}))^T$  o stopniu redundancji 1 spełniającej warunek (3.60). Zdefiniujmy wyróżnik  $\Delta$  konfiguracji osobliwej*

$$\Delta(\mathbf{x}_0) = \det \begin{bmatrix} \frac{\partial^2 H}{\partial q^2} & \frac{\partial^2 H}{\partial q \partial p} \\ \frac{\partial^2 H}{\partial p \partial q} & \frac{\partial^2 H}{\partial p^2} \end{bmatrix} (x_{0m}, x_{0n}).$$

Wówczas, jeżeli wyróżnik

$$\Delta(\mathbf{x}_0) > 0,$$

<sup>†</sup>Zobacz dodatek A.3.

to konfiguracja  $\mathbf{x}_0$  jest lokalnie niemożliwa do uniknięcia. W przypadku, gdy wyróżnik

$$\Delta(\mathbf{x}_0) < 0,$$

konfiguracja  $\mathbf{x}_0$  jest lokalnie możliwa do uniknięcia. W oryginalnych współrzędnych w przestrzeni przegubowej

$$\Delta(\mathbf{x}_0) = AB - C^2,$$

przy czym wyrażenia  $A$ ,  $B$ ,  $C$  są określone przez następujące zależności:

$$\begin{aligned} A &= \frac{\partial^2 k_m}{\partial x_m^2} - \left( \frac{\partial \mathbf{k}^{m-1}}{\partial x_m} \right)^T \mathbf{P} \frac{\partial \mathbf{k}^{m-1}}{\partial x_m} - 2\mathbf{R} \frac{\partial \mathbf{k}^{m-1}}{\partial x_m} - \mathbf{T} \frac{\partial^2 \mathbf{k}^{m-1}}{\partial x_m^2}, \\ B &= \frac{\partial^2 k_m}{\partial x_n^2} - \left( \frac{\partial \mathbf{k}^{m-1}}{\partial x_n} \right)^T \mathbf{P} \frac{\partial \mathbf{k}^{m-1}}{\partial x_n} - 2\mathbf{S} \frac{\partial \mathbf{k}^{m-1}}{\partial x_n} - \mathbf{T} \frac{\partial^2 \mathbf{k}^{m-1}}{\partial x_n^2}, \\ C &= \frac{\partial^2 k_m}{\partial x_m \partial x_n} - \left( \frac{\partial \mathbf{k}^{m-1}}{\partial x_n} \right)^T \mathbf{P} \frac{\partial \mathbf{k}^{m-1}}{\partial x_m} - \mathbf{S} \frac{\partial \mathbf{k}^{m-1}}{\partial x_m} + \\ &\quad - \mathbf{R} \frac{\partial \mathbf{k}^{m-1}}{\partial x_n} - \mathbf{T} \frac{\partial^2 \mathbf{k}^{m-1}}{\partial x_m \partial x_n}, \end{aligned}$$

natomiast macierz  $\mathbf{P}$  i wektory wierszowe  $\mathbf{R}$ ,  $\mathbf{S}$ ,  $\mathbf{T}$  spełniają równania

$$\begin{aligned} \mathbf{T} \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} &= \frac{\partial k_m}{\partial \mathbf{x}^{m-1}}, \\ \left( \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} \right)^T \mathbf{P} \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} + \begin{bmatrix} \mathbf{T} \frac{\partial^2 \mathbf{k}^{m-1}}{\partial x_1 \partial \mathbf{x}^{m-1}} \\ \vdots \\ \mathbf{T} \frac{\partial^2 \mathbf{k}^{m-1}}{\partial x_{m-1} \partial \mathbf{x}^{m-1}} \end{bmatrix} &= \frac{\partial^2 k_m}{\partial \mathbf{x}^{m-1} \partial \mathbf{x}^{m-1}}, \\ \mathbf{R} \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} + \mathbf{T} \frac{\partial^2 \mathbf{k}^{m-1}}{\partial x_m \partial \mathbf{x}^{m-1}} + \left( \frac{\partial \mathbf{k}^{m-1}}{\partial x_m} \right)^T \mathbf{P} \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} &= \frac{\partial^2 k_m}{\partial x_m \partial \mathbf{x}^{m-1}}, \\ \mathbf{S} \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} + \mathbf{T} \frac{\partial^2 \mathbf{k}^{m-1}}{\partial x_n \partial \mathbf{x}^{m-1}} + \left( \frac{\partial \mathbf{k}^{m-1}}{\partial x_n} \right)^T \mathbf{P} \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} &= \frac{\partial^2 k_m}{\partial x_n \partial \mathbf{x}^{m-1}}. \end{aligned}$$

W powyższych wzorach  $\mathbf{k}^{m-1} = (k_1, \dots, k_{m-1})^T$ ,  $\mathbf{x}^{m-1} = (x_1, \dots, x_{m-1})^T$ ,  $n = m + 1$ , a wszystkie pochodne cząstkowe należy obliczyć w konfiguracji osobliwej  $\mathbf{x}_0$ .

### Kinematyka o stopniu redundancji $\geq 1$

Obecnie zajmiemy się kinematyką  $\mathbf{y} = \mathbf{k}(\mathbf{x})$  o dowolnym stopniu redundancji  $r = n - m \geq 1$ . Niech  $\mathbf{x}_0$  oznacza, jak poprzednio, konfigurację osobiwą. Jakobian analityczny  $\mathbf{J}^a(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x})$  ma obecnie rozmiar  $m \times n$ . Ustalmy zbiór wskaźników  $1 \leq i_1 < \dots < i_{m+1} \leq n$  i zdefiniujmy układ dynamiczny

$$\dot{\mathbf{x}} = \mathbf{X}_{i_1 \dots i_{m+1}}(\mathbf{x}), \quad (3.64)$$

którego pole wektorowe  $\mathbf{X}_{i_1 \dots i_{m+1}}(\mathbf{x}) = (X_{i_1 \dots i_{m+1} 1}(\mathbf{x}), \dots, X_{i_1 \dots i_{m+1} n}(\mathbf{x}))^T$  ma składowe

$$X_{i_1 \dots i_{m+1} k}(\mathbf{x}) = \begin{cases} 0, & \text{jeżeli } k \neq i_1, \dots, i_{m+1}, \\ (-1)^{r+1} \det \mathbf{J}_{i_1 \dots i_{m+1}}^{\text{ak}}(\mathbf{x}), & \text{jeżeli } k = i_r \text{ dla pewnego } r. \end{cases} \quad (3.65)$$

W wyrażeniu (3.65)  $\mathbf{J}_{i_1 \dots i_{m+1}}^{\text{ak}}(\mathbf{x})$  oznacza macierz rozmiaru  $m \times m$  złożoną z kolumn jakobianu analitycznego  $\mathbf{J}^a(\mathbf{x})$  o numerach  $i_1, \dots, i_{m+1}$ , z usuniętą kolumną numer  $k$ . Biorąc wszystkie kombinacje wskaźników  $i_1, \dots, i_{m+1}$  uzyskujemy rodzinę układów dynamicznych postaci (3.64), zwanych *układami hamiltonowskimi* stowarzyszonymi z kinematyką  $\mathbf{k}$ . Pola  $\mathbf{X}_{i_1 \dots i_{m+1}}$  noszą nazwę hamiltonowskich. Podobnie, jak w przypadku stopnia redundancji  $r = 1$ , pola hamiltonowskie  $\mathbf{X}_{i_1 \dots i_{m+1}}(\mathbf{x}) \in \text{Ker } \mathbf{J}^a(\mathbf{x})^\ddagger$ , ich dywergencja  $\text{div } \mathbf{X}_{i_1 \dots i_{m+1}}(\mathbf{x}) = 0$ , a konfiguracja osobiwa  $\mathbf{x}_0$  jest punktem równowagi każdego układu hamiltonowskiego. Możliwość lub niemożliwość uniknięcia konfiguracji  $\mathbf{x}_0$  będzie zależna od przebiegu trajektorii układów hamiltonowskich w pobliżu punktu równowagi. Pociąga to za sobą następujący warunek wystarczający na to, aby konfiguracja osobiwa była możliwa do uniknięcia: jest tak, jeżeli istnieje taki układ hamiltonowski  $\mathbf{X}_{i_1 \dots i_{m+1}}(\mathbf{x})$ , którego macierz przybliżenia liniowego w punkcie  $\mathbf{x}_0$  posiada wartość własną o niezerowej części rzeczywistej.

Skoncentrujemy teraz naszą uwagę na konfiguracjach osobiwych korzędu 1. Dla każdej konfiguracji korzędu 1 istnieją takie współrzędne w przestrzeni przegubowej, określone w pewnym otoczeniu konfiguracji osobiwej, że układy hamiltonowskie (3.64) przyjmują szczególnie prostą postać.

<sup>‡</sup>Zauważmy, że pola hamiltonowskie należą do dystrybucji rozpiętej przez kolumny macierzy  $\mathbf{J}_W^{\text{a\#}} \mathbf{J}^a - \mathbb{I}_n$ .

Założmy, że w konfiguracji osobliwej  $\mathbf{x}_0$  jacobian analityczny manipulatora  $\mathbf{J}^a(\mathbf{x}_0) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}_0)$  spełnia warunek

$$\text{rank} \begin{bmatrix} \frac{\partial k_1}{\partial x_1} & \dots & \frac{\partial k_1}{\partial x_{m-1}} \\ \vdots & \ddots & \vdots \\ \frac{\partial k_{m-1}}{\partial x_1} & \dots & \frac{\partial k_{m-1}}{\partial x_{m-1}} \end{bmatrix}(\mathbf{x}_0) = m - 1. \quad (3.66)$$

Korzystając z (3.66), zdefiniujemy wokół  $\mathbf{x}_0$  układ współrzędnych

$$\boldsymbol{\xi} = (k_1(\mathbf{x}), \dots, k_{m-1}(\mathbf{x}), x_m, \dots, x_n)^T.$$

Kinematykę  $\mathbf{y} = \mathbf{k}(\mathbf{x})$  w nowych współrzędnych można zapisać jako

$$\mathbf{y} = \mathbf{k} \circ \boldsymbol{\varphi}^{-1}(\boldsymbol{\xi}) = \mathbf{h}(\boldsymbol{\xi}) = (\xi_1, \dots, \xi_{m-1}, h_m(\boldsymbol{\xi}))^T,$$

przy czym funkcja  $h_m(\boldsymbol{\xi})$  spełnia warunek  $h_m \circ \boldsymbol{\varphi}(\mathbf{x}) = k_m(\mathbf{x})$ . W nowych współrzędnych tylko pola  $\mathbf{X}_{12\dots m m+i}(\boldsymbol{\xi})$ ,  $i = 1, 2, \dots, n - m$ , nie są tożsamościowo równe zeru, przy czym

$$\begin{cases} \mathbf{X}_{12\dots m m+i, m}(\boldsymbol{\xi}) = (-1)^{m+1} \frac{\partial h_m(\boldsymbol{\xi})}{\partial \xi_{m+i}} \\ \mathbf{X}_{12\dots m m+i, m+i}(\boldsymbol{\xi}) = (-1)^{m+2} \frac{\partial h_m(\boldsymbol{\xi})}{\partial \xi_m}, \end{cases} \quad (3.67)$$

natomiast pozostałe składowe pola  $\mathbf{X}_{12\dots m m+i}(\boldsymbol{\xi})$  znikają. Z definicji (3.67) wynika istnienie zbioru niezmienniczego

$$\mathbf{M} = \{ \boldsymbol{\xi} = (\xi_1, \dots, \xi_n)^T \in \mathbb{R}^n \mid \xi_j = k_j(\mathbf{x}_0), j = 1, \dots, m - 1 \},$$

wspólnego dla wszystkich pól hamiltonowskich. Oznaczmy współrzędne na zbiorze  $\mathbf{M} \cong \mathbb{R}^{n-m+1}$  jako  $\boldsymbol{\theta} = (\theta_0, \dots, \theta_{n-m})^T$  i niech  $\mathbf{X}_{12\dots m m+i}^{\mathbf{M}}$  będzie ograniczeniem pola  $\mathbf{X}_{12\dots m m+i}$ ,  $i = 1, 2, \dots, n - m$ , do zbioru niezmienniczego  $\mathbf{M}$ . Zdefiniujemy funkcję

$$H(\boldsymbol{\theta}) = h_m(k_1(\mathbf{x}_0), \dots, k_{m-1}(\mathbf{x}_0), \theta_0, \theta_1, \dots, \theta_{n-m}). \quad (3.68)$$

Nietrudno pokazać, że

$$\begin{aligned} \mathbf{X}_{12\dots m m+i, 0}^{\mathbf{M}}(\boldsymbol{\theta}) &= (-1)^{m+1} \frac{\partial H(\boldsymbol{\theta})}{\partial \theta_i}, \\ \mathbf{X}_{12\dots m m+i, i}^{\mathbf{M}}(\boldsymbol{\theta}) &= (-1)^{m+2} \frac{\partial H(\boldsymbol{\theta})}{\partial \theta_0}. \end{aligned} \quad (3.69)$$

Co więcej, funkcja  $H(\boldsymbol{\theta})$  jest całką pierwszą (stałą ruchu)<sup>§</sup> każdego z pól wektorowych (3.69), co oznacza, że

$$dH\mathbf{X}_{12\dots m m+i}^M(\boldsymbol{\theta}) = 0.$$

Bezpośrednią konsekwencją istnienia stałej ruchu jest następujący rezultat.

**Twierdzenie 3.2.2** *Niech  $\mathbf{x}_0 = (x_{01}, x_{02}, \dots, x_{0m-1}, x_{0m}, \dots, x_{0n})^T \in \mathbb{R}^n$  będzie konfiguracją osobliwą rzędu 1 kinematyki manipulatora  $\mathbf{k}(\mathbf{x}) = (k_1(\mathbf{x}), k_2(\mathbf{x}), \dots, k_m(\mathbf{x}))^T$  o stopniu redundancji  $\geq 1$  spełniającej warunek (3.66). Załóżmy, że macierz Hessego stałej ruchu*

$$\mathbf{H}(\mathbf{x}_0) = \frac{\partial^2 H}{\partial \boldsymbol{\theta}^2}(x_{0m}, x_{0m+1}, \dots, x_{0n}) \quad (3.70)$$

*ma niezerowe wartości własne i określmy indeks  $\Gamma(\mathbf{x}_0)$  macierzy  $\mathbf{H}(\mathbf{x}_0)$  jako liczbę ujemnych wartości własnych  $\mathbf{H}(\mathbf{x}_0)$ . Wówczas, jeżeli*

$$\Gamma(\mathbf{x}_0) = 0 \quad \text{lub} \quad \Gamma(\mathbf{x}_0) = n - m + 1,$$

*to  $\mathbf{x}_0$  jest konfiguracją lokalnie niemożliwą do uniknięcia. Jeżeli*

$$1 \leq \Gamma(\mathbf{x}_0) \leq n - m,$$

*to konfiguracja  $\mathbf{x}_0$  jest lokalnie możliwa do uniknięcia. W oryginalnych współrzędnych w przestrzeni przegubowej, macierz (3.70) można wyrazić w następującej formie*

$$\mathbf{H}(\mathbf{x}_0) = -\mathbf{M}^T(\mathbf{X} - \mathbf{P})\mathbf{M} - \left( \mathbf{M}^T(\mathbf{Y} - \mathbf{Q}) + (\mathbf{M}^T(\mathbf{Y} - \mathbf{Q}))^T \right) + (\mathbf{W} - \mathbf{S}),$$

gdzie

$$\begin{aligned} \mathbf{M} &= \left( \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} \right)^{-1} \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}_m}, & \mathbf{X} &= \frac{\partial^2 \mathbf{k}_m}{\partial (\mathbf{x}^{m-1})^2}, \\ \mathbf{Y} &= \frac{\partial^2 \mathbf{k}_m}{\partial \mathbf{x}^{m-1} \partial \mathbf{x}_m}, & \mathbf{W} &= \frac{\partial^2 \mathbf{k}_m}{\partial \mathbf{x}_m^2}, \\ P_{ij} &= \frac{\partial \mathbf{k}_m}{\partial \mathbf{x}^{m-1}} \left( \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} \right)^{-1} \frac{\partial^2 \mathbf{k}^{m-1}}{\partial x_i \partial x_j}, & i, j &\leq m-1, \end{aligned}$$

<sup>§</sup>Zobacz dodatek A.3.

$$Q_{ij} = \frac{\partial k_m}{\partial \mathbf{x}^{m-1}} \left( \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} \right)^{-1} \frac{\partial^2 \mathbf{k}^{m-1}}{\partial x_i \partial x_j}, \quad i \leq m-1, \quad m \leq j \leq n,$$

$$S_{ij} = \frac{\partial k_m}{\partial \mathbf{x}^{m-1}} \left( \frac{\partial \mathbf{k}^{m-1}}{\partial \mathbf{x}^{m-1}} \right)^{-1} \frac{\partial^2 \mathbf{k}^{m-1}}{\partial x_i \partial x_j}, \quad m \leq i, \quad j \leq n,$$

przy oznaczeniach  $\mathbf{x}^{m-1} = (x_1, \dots, x_{m-1})^T$ ,  $\mathbf{x}_m = (x_m, \dots, x_n)^T$ ,  $\mathbf{k}^{m-1} = (k_1, \dots, k_{m-1})^T$  oraz przy założeniu, że wszystkie pochodne cząstkowe zostały obliczone w konfiguracji osobliwej  $\mathbf{x}_0$ <sup>¶</sup>.

### Przykład 3.2.1 (Manipulator IRb-6 na torze jezdny)

Dla ilustracji warunków unikania konfiguracji osobliwych podanych w twierdzeniu 3.2.2 rozważymy kinematykę manipulatora IRb-6 zamontowanego na torze jezdny w aspekcie współrzędnych położenia,

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} c_2 r + e(c_2 c_5 c_6 + s_2 s_6) \\ x_1 + s_2 r + e(s_2 c_5 c_6 - c_2 s_6) \\ -d_1 + d_6 c_5 - a_2 s_3 - a_3 s_4 - e c_6 s_5 \end{pmatrix}, \quad (3.71)$$

z efektem przesuniętym o pewną odległość  $e$ . W równaniu (3.71) parametr  $r = a_2 c_3 + a_3 c_4 + d_6 s_5$ . Współrzędne przegubowe i zadaniowe zostały ponumerowane w taki sposób, żeby był spełniony warunek (3.66). Wybierzmy konfigurację osobliwą  $\mathbf{x}_0 = (x_1, x_2, \pi/2, \pi/2, 0, \pi/2)^T$ . Po wykonaniu odpowiednich obliczeń stwierdzamy, że macierz  $\mathbf{H}(\mathbf{x}_0)$  w tej konfiguracji ma postać

$$\mathbf{H}(\mathbf{x}_0) = \begin{bmatrix} a_2 & 0 & 0 & 0 \\ 0 & a_3 & 0 & 0 \\ 0 & 0 & -d_6 & e \\ 0 & 0 & e & 0 \end{bmatrix}. \quad (3.72)$$

Łatwo sprawdzić, że macierz  $\mathbf{H}(\mathbf{x}_0)$  ma jedną ujemną wartość własną, zatem jej indeks  $\Gamma(\mathbf{x}_0) = 1$ . Wynika stąd, na mocy twierdzenia 3.2.2, że wybrana konfiguracja osobliwa jest możliwa do uniknięcia. ■

<sup>¶</sup>W przypadku, gdy stopień redundancji  $r = 1$ , wyznacznik macierzy  $\mathbf{H}(\mathbf{x}_0)$  jest równy wyróżnikowi  $\Delta(\mathbf{x}_0)$ . Wyróżnik  $\Delta(\mathbf{x}_0) > 0$  oznacza, że  $\mathbf{H}(\mathbf{x}_0)$  ma obie wartości własne jednakowego znaku,  $\Delta(\mathbf{x}_0) < 0$  — że wartości własne są przeciwnych znaków. W rezultacie, twierdzenie 3.2.1 staje się szczególnym przypadkiem twierdzenia 3.2.2.

### 3.2.2 Metoda postaci normalnych

#### Pojęcia podstawowe

Zasada unikania konfiguracji osobliwych przy rozwiązywaniu odwrotnego zadania kinematyki może być zastosowana do manipulatorów redundantnych wyłącznie w odniesieniu do tych konfiguracji osobliwych, które są możliwe do uniknięcia. Wszelako jednak, dla manipulatorów nieredundantnych lub dla niemożliwych do uniknięcia konfiguracji osobliwych manipulatorów redundantnych, potrzebne są specjalne algorytmy kinematyki odwrotnej, zdolne do wyznaczenia trajektorii ruchu przegubów przechodzących przez konfiguracje osobliwe. Takie algorytmy będziemy nazywać *algorytmami osobliwej kinematyki odwrotnej*. Przypomnijmy sformułowanie odwrotnego zadania kinematyki manipulatora we współrzędnych:

Mając daną kinematykę manipulatora  $\mathbf{y} = \mathbf{k}(\mathbf{x})$  i trajektorię zadaniową  $\mathbf{y}_d(t)$ ,  $t \in \mathcal{I}$ , wyznaczyć trajektorię przegubową  $\mathbf{x}_d(t)$ , taką że  $\mathbf{y}_d(t) = \mathbf{k}(\mathbf{x}_d(t))$ .

Przy rozwiązywaniu osobliwego odwrotnego zadania kinematyki dopuszczamy, aby trajektoria przegubowa przechodziła przez lub w pobliżu konfiguracji osobliwych manipulatora. Do metod pozwalających na uzyskanie rozwiązania tego zadania w postaci jawnej należy metoda postaci normalnych. Jej główna idea polega na przekształceniu odwrotnego zadania kinematyki dla kinematyki oryginalnej do zadania dla reprezentującej ją równoważnej postaci normalnej, rozwiązaniu zadania dla postaci normalnej i przekształceniu uzyskanego rozwiązania z powrotem do współrzędnych kinematyki oryginalnej.

Zdefiniowanie postaci normalnej kinematyki wymaga wprowadzenia pojęcia równoważności odwzorowań, zgodnie z którym dwa odwzorowania są *równoważne*, jeżeli stają się identyczne w odpowiednio dobranych układach współrzędnych. Bardziej formalnie, gładkie odwzorowania  $\mathbf{f}, \mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^m$  nazywamy równoważnymi, jeżeli istnieją układy współrzędnych (dyfeomorfizmy)  $\boldsymbol{\varphi} : \mathbb{R}^n \rightarrow \mathbb{R}^n$  i  $\boldsymbol{\psi} : \mathbb{R}^m \rightarrow \mathbb{R}^m$ , takie że diagram

$$\begin{array}{ccc} \mathbb{R}^n & \xrightarrow{\mathbf{f}} & \mathbb{R}^m \\ \boldsymbol{\varphi} \downarrow & & \downarrow \boldsymbol{\psi} \\ \mathbb{R}^n & \xrightarrow{\mathbf{g}} & \mathbb{R}^m \end{array}$$

jest przemienny, tzn.  $\boldsymbol{\psi} \circ \mathbf{f} = \mathbf{g} \circ \boldsymbol{\varphi}$ . Równoważność odwzorowań jest lo-



kalna, jeżeli układy współrzędnych  $\boldsymbol{\varphi}$ ,  $\boldsymbol{\psi}$  są określone lokalnie. Używana przez nas równoważność wywodzi się z teorii osobliwości odwzorowań i nosi nazwę *RL-równoważności* (right-left). Klasa odwzorowań równoważnych danemu odwzorowaniu  $\mathbf{f}$  może być reprezentowana przez odpowiednio wybranego przedstawiciela tej klasy, nazywanego *postacią normalną* odwzorowania  $\mathbf{f}$ . Zwykle wymaga się, aby postać normalna była „prosta” z estetycznego punktu widzenia\*. W zagadnieniach kinematyki manipulatorów postać normalna powinna przede wszystkim umożliwiać rozwiązanie odwrotnego zadania kinematyki.

Niech  $\mathbf{k} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ ,  $\mathbf{y} = \mathbf{k}(\mathbf{x})$ ,  $n \geq m$  oznacza kinematykę manipulatora. Załóżmy, że  $\mathbf{x}_0 \in \mathbb{R}^n$  jest konfiguracją regularną, tzn. taką że  $\text{rank } \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}_0) = m$ . Zdefiniujmy lokalne układy współrzędnych

$$\begin{aligned}\boldsymbol{\xi} &= \boldsymbol{\varphi}(\mathbf{x}) = (\mathbf{k}(\mathbf{x}) - \mathbf{k}(\mathbf{x}_0), x_{m+1}, \dots, x_n)^T, \\ \boldsymbol{\eta} &= \boldsymbol{\psi}(\mathbf{y}) = \mathbf{y} - \mathbf{k}(\mathbf{x}_0).\end{aligned}\tag{3.73}$$

Zgodnie z twierdzeniem o funkcji odwrotnej, odwzorowanie  $\boldsymbol{\varphi}$  jest lokalnym dyfeomorfizmem przekształcającym pewne otoczenie konfiguracji  $\mathbf{x}_0$  na pewne otoczenie  $\mathbf{0} \in \mathbb{R}^n$ . Analogicznie,  $\boldsymbol{\psi}$  jest lokalnym dyfeomorfizmem odwzorowującym otoczenie  $\mathbf{y}_0 = \mathbf{k}(\mathbf{x}_0)$  na otoczenie  $\mathbf{0} \in \mathbb{R}^m$ . Łatwo zauważyć, że dyfeomorfizmy (3.73) przekształcają kinematykę  $\mathbf{k}(\mathbf{x})$  do postaci

$$\boldsymbol{\eta} = \mathbf{k}_0(\boldsymbol{\xi}) = \boldsymbol{\psi} \circ \mathbf{k} \circ \boldsymbol{\varphi}^{-1}(\boldsymbol{\xi}) = (\xi_1, \dots, \xi_m).\tag{3.74}$$

Jak widzimy, przy użyciu dyfeomorfizmów (3.73) jesteśmy w stanie sprowadzić kinematykę nieosobliwą do postaci liniowej projekcji  $\mathbb{R}^n$  na  $\mathbb{R}^m$ <sup>†</sup> będącej *postacią normalną* kinematyki nieosobliwej. Ponieważ powyższa elementarna obserwacja jest istotna z punktu widzenia odwrotnego zadania kinematyki i zadania śledzenia trajektorii, sformułujemy ją formalnie.

**Wniosek 3.2.1** *Założmy, że  $\mathbf{x}_0$  jest konfiguracją regularną kinematyki  $\mathbf{k} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ ,  $\mathbf{y} = \mathbf{k}(\mathbf{x})$ . Wówczas, kinematyka ta jest lokalnie*

\* Jak zaznaczyliśmy we *Wprowadzeniu*, za patrona metody postaci normalnych obie-ramy W. Ockhama.

<sup>†</sup> W rozumieniu teorii osobliwości kinematyka  $\mathbf{k}(\mathbf{x})$  jest lokalnie RL-równoważna postaci (3.74). W dalszej części niniejszego podrozdziału będziemy mówić, że kinematyka jest lokalnie równoważna pewnej postaci normalnej lub że można ją przekształcić do pewnej postaci normalnej.

równoważna liniowej projekcji

$$\mathbf{k}_0(x_1, \dots, x_n) = (x_1, \dots, x_m)^T. \quad (3.75)$$

Używając języka postaci normalnych powiemy, że algorytmy kinematyki odwrotnej manipulatorów opisane w podrozdziale 3.1 mogą być stosowane jedynie do kinematyki mającej postać normalną typu liniowej projekcji. W tym miejscu pojawia się pytanie: W jaki sposób rozwiązać odwrotne zadanie kinematyki w przypadku kinematyki, która nie jest równoważna postaci (3.75)?

### Algorytm

Metodą pozwalającą na rozwiązanie tak postawionego zadania jest *metoda postaci normalnych*. Zgodnie z przesłaniem metody, rozwiązanie odwrotnego zadania kinematyki sformułowanego dla kinematyki oryginalnej sprowadzone zostaje do rozwiązania odwrotnego zadania kinematyki dla postaci normalnej kinematyki. Przy założeniu, że trajektoria zadana przecina osobliwości w dyskretnych chwilach czasu, dyrektywy postępowania wynikające z metody postaci normalnych można sformułować w postaci następującej procedury.

- Krok 1.** Dla kinematyki określonej we współrzędnych  $\mathbf{k} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ ,  $\mathbf{y} = \mathbf{k}(\mathbf{x})$  i trajektorii zadanej  $\mathbf{y}_d(t)$ ,  $t \in \mathcal{I} = [t_p, t_k]$  znaleźć zbiór konfiguracji osobliwych  $S$  i wyznaczyć uporządkowany zbiór chwil  $\{t_{si}\} = T$ ,  $i > 0$ , w których trajektoria  $\mathbf{y}_d(t)$  przecina zbiór wartości osobliwych  $\mathbf{k}(S)$  kinematyki  $\mathbf{k}$ . Przyjmą  $t_0 = t_p$ ,  $i = 1$ . Jeśli chwila  $t_0$  należy do otoczenia  $t_{s1}$  przejść do kroku 3.
- Krok 2.** Zastosować do rozwiązania zadania kinematyki odwrotnej dla czasów  $t \in [t_{i-1}, t_{si} - \varepsilon_i]$ , przy wybranym  $\varepsilon_i > 0$ , jedną z metod pozwalających na rozwiązanie regularnego odwrotnego zadania kinematyki, opisanych w podrozdziale 3.1, z warunkiem początkowym  $\mathbf{x}_d(t_{i-1})$ , takim że  $\mathbf{y}_d(t_{i-1}) = \mathbf{k}(\mathbf{x}_d(t_{i-1}))$ .
- Krok 3.** Określić konfigurację osobliwą  $\mathbf{x}_{si} = \mathbf{x}(t_{si})$ , wyznaczyć postać normalną  $\mathbf{k}_0$  kinematyki  $\mathbf{k}$  w otoczeniu tej konfiguracji oraz znaleźć układy współrzędnych  $\boldsymbol{\varphi} : \mathbf{U} \rightarrow \boldsymbol{\varphi}(\mathbf{U})$ ,  $\boldsymbol{\xi} = \boldsymbol{\varphi}(\mathbf{x})$  i  $\boldsymbol{\psi} : \mathbf{V} \rightarrow \boldsymbol{\psi}(\mathbf{V})$ ,  $\boldsymbol{\eta} = \boldsymbol{\psi}(\mathbf{y})$  określone na otwartych otoczeniach  $\mathbf{U} \ni \mathbf{x}_{si}$  i  $\mathbf{V} \ni \mathbf{k}(\mathbf{x}_{si})$ , takie że

$$\mathbf{k}_0 \circ \boldsymbol{\varphi}(\mathbf{x}) = \boldsymbol{\psi} \circ \mathbf{k}(\mathbf{x}). \quad (3.76)$$

**Krok 4.** Wyrazić trajektorię zadaną  $\mathbf{y}_d(t) \in \mathbf{V}$  (dla  $t \in [t_{si} - \varepsilon_i, t_{si} + \varepsilon_i]$ ) we współrzędnych  $\boldsymbol{\eta}$ , zgodnie z zależnością

$$\boldsymbol{\eta}_d(t) = \boldsymbol{\psi}(\mathbf{y}_d(t)).$$

**Krok 5.** Rozwiązać odwrotne zadanie kinematyki dla postaci normalnej  $\mathbf{k}_0$  i trajektorii zadanej  $\boldsymbol{\eta}_d(t)$ , tzn. znaleźć  $\boldsymbol{\xi}_d(t) \in \varphi(\mathbf{U})$ , które spełnia równanie

$$\boldsymbol{\eta}_d(t) = \mathbf{k}_0(\boldsymbol{\xi}_d(t)).$$

**Krok 6.** Przekształcić uzyskane rozwiązanie  $\boldsymbol{\xi}_d(t)$  z powrotem do oryginalnych współrzędnych przegubowych obliczając

$$\boldsymbol{x}_d(t) = \boldsymbol{\varphi}^{-1}(\boldsymbol{\xi}_d(t)) \in \mathbf{U}.$$

**Krok 7.** Przyjąć  $t_i = t_{si} + \varepsilon_i$ , a następnie  $i \leftarrow i + 1$ . Jeśli  $t_{si} \in T$  powtórzyć kroki 2–6. W przeciwnym razie, jeśli  $t_{i-1} < t_k$ , wykonać krok 2 dla chwil czasu  $t \in [t_{i-1}, t_k]$ . W wyniku połączenia rozwiązania regularnego (otrzymanego w kroku 2) i osobliwego (z kroku 6) uzyskujemy kompletne rozwiązanie odwrotnego zadania kinematyki w postaci trajektorii  $\boldsymbol{x}_d(t)$ ,  $t \in [t_p, t_k]$  określonej w przestrzeni przegubowej manipulatora.

Jak widać, algorytm metody postaci normalnych składa się z dwóch zasadniczych części: pierwszej, pozwalającej na wyznaczenie odcinków trajektorii przebiegających w otoczeniu konfiguracji regularnych (krok 2), i drugiej, w której są określane odcinki trajektorii położone w otoczeniu konfiguracji osobliwych (kroki 3–6). Rozwiązanie osobliwego odwrotnego zadania kinematyki polega więc na sukcesywnym wyznaczaniu nieosobliwych i osobliwych fragmentów trajektorii przegubowej, a następnie ich połączeniu w fazie końcowej (krok 7). O długości fragmentów osobliwych, określonej parametrami  $\varepsilon_i$ , a co za tym idzie o doborze wartości tych parametrów, decydują możliwości metody rozwiązania regularnego zadania kinematyki odwrotnej, zastosowanej w kroku 2 algorytmu, oraz wielkość dziedziny dyfeomorfizmów  $\boldsymbol{\varphi}$  i  $\boldsymbol{\psi}$  wyznaczonych w kroku 3. W skrajnych przypadkach, cała poszukiwana trajektoria może być wyznaczona wyłącznie z wykorzystaniem albo osobliwej (patrz przykład 3.2.3), albo nieosobliwej (patrz przykłady 3.2.4 i 3.2.6) części algorytmu.

Analiza algorytmu metody postaci normalnych prowadzi do wniosku, że najtrudniejsze do realizacji są kroki 3 i 5. W kroku 3 musimy określić postać normalną kinematyki w konfiguracji osobliwej oraz znaleźć postacie dyfeomorfizmów transformujących kinematykę do postaci normalnej. Praktyczny sposób znajdowania dyfeomorfizmów zostanie przedstawiony nieco dalej, na przykładzie manipulatora typu podwójne wahadło (przykład 3.2.3)<sup>‡</sup>. Drugim trudnym do przejścia etapem algorytmu jest krok 5. Postać normalna kinematyki musi być wystarczająco prosta, aby było dla niej możliwe rozwiązanie odwrotnego zadania kinematyki. Dyskusją tego zagadnienia zajmiemy się w dalszej części niniejszego podrozdziału, po wyznaczeniu typowych postaci normalnych kinematyki.

Obecnie przedstawimy sposoby określania postaci normalnych kinematyki. Zgodnie z rozważaniami przeprowadzonymi w rozdziale 2.3.5, konfiguracje korzędu 1, należące do zbioru  $S_1$ , tworzą największy podzbiór konfiguracji osobliwych, często niemożliwych do uniknięcia podczas planowania i śledzenia trajektorii. Z tego względu skupimy uwagę na analizie kinematyki manipulatorów w otoczeniu konfiguracji należących do  $S_1$ . Naszym celem będzie podanie warunków, które pozwolą na zdefiniowanie postaci normalnych kinematyki w otoczeniu konfiguracji osobliwych korzędu 1.

## Postacie normalne

Rozpatrzmy reprezentację kinematyki we współrzędnych, w formie odwzorowania analitycznego

$$\mathbf{k}: \mathbb{R}^n \longrightarrow \mathbb{R}^m, \quad \mathbf{y} = \mathbf{k}(\mathbf{x}) = (k_1(\mathbf{x}), \dots, k_m(\mathbf{x}))^T. \quad (3.77)$$

Oznaczmy przez  $\mathbf{x}_0 \in S_1$  konfigurację osobliwą korzędu 1 kinematyki. Dla  $\mathbf{x}_0$  mamy

$$\text{rank} \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x}_0) = \text{rank} \mathbf{J}^a(\mathbf{x}_0) = m - 1, \quad (3.78)$$

gdzie  $\mathbf{J}^a(\mathbf{x})$  oznacza jacobian analityczny manipulatora. Zauważmy, że dla kinematyki  $\mathbf{y} = \mathbf{k}(\mathbf{x})$  i konfiguracji osobliwej  $\mathbf{x}_0 \in S_1$ , po zastosowaniu odpowiedniej permutacji współrzędnych  $(x_1, \dots, x_n)$  i  $(y_1, \dots, y_m)$ , warun-

<sup>‡</sup>Pokrótcie, wykorzystuje się w tym celu własność, że dyfeomorfizmy  $\varphi$  i  $\psi$  są analityczne, mogą więc być reprezentowane przez ich rozwinięcia w szereg Taylora. Jesteśmy w ten sposób w stanie wyznaczyć je z dowolnie dużą dokładnością.

kowi (3.78) można nadać postać

$$\text{rank} \begin{bmatrix} \frac{\partial k_1}{\partial x_1} & \dots & \frac{\partial k_1}{\partial x_{m-1}} \\ \vdots & \ddots & \vdots \\ \frac{\partial k_{m-1}}{\partial x_1} & \dots & \frac{\partial k_{m-1}}{\partial x_{m-1}} \end{bmatrix} (\mathbf{x}_0) = m - 1. \quad (3.79)$$

Zdefiniujmy dla macierzy  $\mathbf{J}^a(\mathbf{x})$  zbiór  $n - m + 1$  podmacierzy  $\mathbf{J}^{ai}(\mathbf{x})$ ,  $i = m, m + 1, \dots, n$  rozmiaru  $m \times m$  złożonych z  $m - 1$  pierwszych kolumn  $\mathbf{J}^a(\mathbf{x})$  oraz kolumny  $i$ -tej tej macierzy. Oznaczmy przez  $\mathbf{v}_m, \mathbf{v}_{m+1}, \dots, \mathbf{v}_n$  bazę przestrzeni  $\text{Ker } \mathbf{J}^a(\mathbf{x}_0)$ . Warunek równoważności kinematyki kwadratowej postaci normalnej przedstawia następujące twierdzenie.

**Twierdzenie 3.2.3** *Niech  $\mathbf{x}_0$  będzie konfiguracją osobliwą korzędu 1 kinematyki  $\mathbf{k}(\mathbf{x})$  spełniającej warunek (3.79). Wówczas, jeżeli*

$$\text{rank} \begin{bmatrix} d(\det \mathbf{J}^{am}) \\ d(\det \mathbf{J}^{am+1}) \\ \vdots \\ d(\det \mathbf{J}^{an}) \end{bmatrix} (\mathbf{x}_0) [\mathbf{v}_m \cdots \mathbf{v}_n] = n - m + 1, \quad (3.80)$$

to w pewnym otoczeniu konfiguracji  $\mathbf{x}_0$  kinematyka  $\mathbf{k}(\mathbf{x})$  jest równoważna kwadratowej postaci normalnej

$$\mathbf{k}_0(\mathbf{x}) = (x_1, x_2, \dots, x_{m-1}, q(x_m, \dots, x_n))^T, \quad (3.81)$$

gdzie  $q(x_m, \dots, x_n) = -x_m^2 - \dots - x_{m-1+s}^2 + x_{m+s}^2 + \dots + x_n^2$ , dla pewnego  $s = 0, \dots, n - m + 1$ .

Nietrudno sprawdzić, że warunek rzędu (3.80) nie zależy od przyjętych układów współrzędnych przegubowych i zadaniowych.

Dla kinematyki nieredundantnej, z podanego twierdzenia można wyciągnąć następujący wniosek.

**Wniosek 3.2.2** *Niech  $\mathbf{x}_0$  będzie konfiguracją osobliwą korzędu 1 kinematyki nieredundantnej  $\mathbf{k} : \mathbb{R}^n \rightarrow \mathbb{R}^n$  spełniającej warunek (3.79) przy  $m = n$ . Jeżeli*

$$d(\det \mathbf{J}^a)(\mathbf{x}_0) \mathbf{v} \neq 0, \quad (3.82)$$

gdzie wektor  $\mathbf{v}$  jest rozwiązaniem równania  $\mathbf{J}^a(\mathbf{x}_0) \mathbf{v} = 0$ , to w pewnym

otoczeniu konfiguracji  $\mathbf{x}_0$  kinematyka  $\mathbf{k}(\mathbf{x})$  jest równoważna kwadratowej (parabolicznej) postaci normalnej

$$\mathbf{k}_0(\mathbf{x}) = (x_1, x_2, \dots, x_{n-1}, x_n^2)^\top. \quad (3.83)$$

Przeprowadzone dotąd badania konfiguracji osobliwych kinematyki manipulatorów wykazują, że kwadratowa postać normalna jest najpowszechniejszą postacią normalną kinematyki w otoczeniu konfiguracji osobliwych korzędu 1. Mniej rozpowszechnionym typem postaci normalnej jest *hiperboliczna postać normalna*. W celu określenia warunków jej występowania, dla kinematyki (3.77) spełniającej warunek (3.79) zdefiniujemy w otoczeniu konfiguracji osobliwej  $\mathbf{x}_0$  lokalny układ współrzędnych

$$\boldsymbol{\xi} = \boldsymbol{\varphi}(\mathbf{x}) = (k_1(\mathbf{x}), \dots, k_{m-1}(\mathbf{x}), x_m, \dots, x_n)^\top, \quad (3.84)$$

przekształcający otoczenie  $\mathbf{U}$  punktu  $\mathbf{x}_0$  w otoczenie  $\mathbf{V}$  punktu  $\boldsymbol{\xi}_0 = \boldsymbol{\varphi}(\mathbf{x}_0)$ . Nazwijmy przekrojem otoczenia  $\mathbf{V}$  zbiór

$$\mathbf{V}_{l_1 \dots l_r} = \left\{ \boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_n)^\top \in \mathbf{V} \mid \begin{array}{l} \xi_{l_1} = \xi_{0l_1}, \\ \xi_{l_2} = \xi_{0l_2}, \dots, \xi_{l_r} = \xi_{0l_r} \end{array} \right\}.$$

Utwórzmy zbiór  $n - m + 1$  podmacierzy  $\mathbf{J}^{a_i}(\mathbf{x})$ ,  $i = m, m + 1, \dots, n$ , określonych tak samo, jak poprzednim twierdzeniu. Warunki istnienia hiperbolicznej postaci normalnej kinematyki podaje następujące twierdzenie.

**Twierdzenie 3.2.4** *Niech  $\mathbf{x}_0$  będzie konfiguracją osobliwą korzędu 1 kinematyki  $\mathbf{k}(\mathbf{x})$  spełniającej warunek (3.79) i niech  $\boldsymbol{\varphi}(\mathbf{x})$  będzie zdefiniowane przez (3.84). Jeżeli istnieją wskaźniki  $i_1, \dots, i_r \in \{m, m + 1, \dots, n\}$ ,  $r = n - m + 1$ , wraz z odpowiadającymi im wskaźnikami  $j_1, \dots, j_r \in \{1, 2, \dots, m - 1\}$ , takie że*

$$\left\{ \begin{array}{l} \det \mathbf{J}^{a_{i_1}} \circ \boldsymbol{\varphi}^{-1} \Big|_{\mathbf{V}} \neq 0 \wedge \det \mathbf{J}^{a_{i_1}} \circ \boldsymbol{\varphi}^{-1} \Big|_{\mathbf{V}_{j_1}} \equiv 0 \\ \det \mathbf{J}^{a_{i_2}} \circ \boldsymbol{\varphi}^{-1} \Big|_{\mathbf{V}_{i_1}} \neq 0 \wedge \det \mathbf{J}^{a_{i_2}} \circ \boldsymbol{\varphi}^{-1} \Big|_{\mathbf{V}_{i_1 j_2}} \equiv 0 \\ \vdots \\ \det \mathbf{J}^{a_{i_r}} \circ \boldsymbol{\varphi}^{-1} \Big|_{\mathbf{V}_{i_1 \dots i_{r-1}}} \neq 0 \wedge \det \mathbf{J}^{a_{i_r}} \circ \boldsymbol{\varphi}^{-1} \Big|_{\mathbf{V}_{i_1 \dots i_{r-1} j_r}} \equiv 0, \end{array} \right.$$

oraz dla  $s = 1, 2, \dots, r$

$$d(\det \mathbf{J}^{\text{ais}})(\mathbf{x}_0) \left( \frac{\partial \boldsymbol{\varphi}}{\partial \mathbf{x}}(\mathbf{x}_0) \right)_{j_s \text{ kol}}^{-1} \neq 0,$$

to w pewnym otoczeniu konfiguracji  $\mathbf{x}_0$  kinematyka  $\mathbf{k}(\mathbf{x})$  jest równoważna hiperbolicznej postaci normalnej

$$\mathbf{k}_0(\mathbf{x}) = (x_1, x_2, \dots, x_{m-1}, x_{i_1}x_{j_1} + \dots + x_{i_r}x_{j_r})^T. \quad (3.85)$$

Podobnie jak w przypadku kwadratowej postaci normalnej, na podstawie powyższego twierdzenia jesteśmy w stanie sformułować wniosek dotyczący występowania hiperbolicznej postaci normalnej kinematyki nieredundantnej.

**Wniosek 3.2.3** Niech  $\mathbf{x}_0$  będzie konfiguracją osobliwą korzędu 1 kinematyki nieredundantnej  $\mathbf{k} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ , spełniającej warunek (3.79) przy  $m = n$ , i niech  $\boldsymbol{\varphi}(\mathbf{x})$  będzie zdefiniowane wzorem (3.84) również dla  $m = n$ . Oznaczmy przez  $S_1$  zbiór konfiguracji osobliwych korzędu 1 kinematyki  $\mathbf{k}(\mathbf{x})$ . Załóżmy, że

$$\det \mathbf{J}^a|_{\mathbf{u}} = \det \mathbf{J}^a \circ \boldsymbol{\varphi}^{-1}|_{\mathbf{v}} \neq 0.$$

Jeżeli istnieje wskaźnik  $j \in \{1, 2, \dots, n-1\}$ , taki że

$$\boldsymbol{\varphi}^{-1}(\mathbf{V}_j) \subset S_1 \cap \mathbf{u} \quad (3.86)$$

oraz

$$d(\det \mathbf{J}^a)(\mathbf{x}_0) \left( \frac{\partial \boldsymbol{\varphi}}{\partial \mathbf{x}}(\mathbf{x}_0) \right)_{j \text{ kol}}^{-1} \neq 0, \quad (3.87)$$

to w pewnym otoczeniu konfiguracji  $\mathbf{x}_0$  kinematyka  $\mathbf{k}(\mathbf{x})$  jest równoważna hiperbolicznej postaci normalnej

$$\mathbf{k}_0(\mathbf{x}) = (x_1, x_2, \dots, x_{n-1}, x_j x_n)^T. \quad (3.88)$$

Przed pokazaniem przykładów zastosowania metody postaci normalnych przedyskutujemy rozwiązanie odwrotnego zadania kinematyki dla parabolicznej i hiperbolicznej postaci normalnej kinematyki nieredundantnej<sup>§</sup>. W przypadku pierwszej z nich, zdefiniowanej jako

$$\mathbf{y} = \mathbf{k}_0(\mathbf{x}) = (x_1, x_2, \dots, x_{n-1}, x_n^2)^T,$$

<sup>§</sup>Zasygnalizowane w kroku 5 algorytmu przedstawionego na stronie 142.

dla trajektorii zadaniowej  $\mathbf{y}_d(t) = (y_{d1}(t), \dots, y_{dn}(t))^T$  zależnej w sposób gładki od czasu i przechodzącej w chwili  $t_s$  przez konfigurację osobliwą (tzn. takiej, że  $y_{dn}(t_s) = 0$ ), otrzymujemy następujące cztery rozwiązania odwrotnego zadania kinematyki

$$\mathbf{x}_d(t) = (y_{d1}(t), \dots, y_{dn-1}(t), \pm \sqrt{y_{dn}(t)})^T, \quad (3.89)$$

$$\mathbf{x}_d(t) = (y_{d1}(t), \dots, y_{dn-1}(t), \pm \operatorname{sgn}(t - t_s) \sqrt{y_{dn}(t)})^T, \quad (3.90)$$

z których jedynie dwa ostatnie są zawsze gładkie. Widzimy, że warunkiem koniecznym istnienia rozwiązania jest spełnienie nierówności  $y_{dn}(t) \geq 0$ .

Dla hiperbolicznej postaci normalnej

$$\mathbf{y} = \mathbf{k}_0(\mathbf{x}) = (x_1, x_2, \dots, x_{n-1}, x_j x_n)^T,$$

$j = 1, \dots, n - 1$ , przy założeniu gładkiej trajektorii zadaniowej  $\mathbf{y}_d(t) = (y_{d1}(t), \dots, y_{dn}(t))^T$  przechodzącej przez konfigurację osobliwą w chwili  $t_s$ <sup>¶</sup>, otrzymujemy rozwiązanie odwrotnego zadania kinematyki

$$\mathbf{x}_d(t) = \mathbf{r}_0(\mathbf{y}_d(t)) = (y_{d1}(t), \dots, y_{dn-1}(t), r_0(t))^T, \quad (3.91)$$

gdzie  $r_0(t) = \frac{y_{dn}(t)}{y_{dj}(t)}$  dla  $t \neq t_s$  oraz  $r_0(t)$  jest nieokreślone dla  $t = t_s$ . Zauważmy, że warunkiem koniecznym istnienia rozwiązania zadania odwrotnego jest  $y_{dn}(t_s) = 0$ . Aby uzyskana wówczas trajektoria  $\mathbf{x}_d(t)$  była ciągła, spośród możliwych wartości  $r_0(t_s)$  należy wybrać  $r_0(t_s) = \lim_{\tau \rightarrow t_s} \frac{y_{dn}(\tau)}{y_{dj}(\tau)}$ .

Obecnie zilustrujemy na przykładach sposób określenia postaci normalnych kinematyki w otoczeniu konfiguracji osobliwych oraz zastosowanie metody postaci normalnych do rozwiązania osobliwego odwrotnego zadania kinematyki.

### Przykład 3.2.2 (Manipulator typu podwójne wahadło)

Manipulator typu podwójne wahadło może zostać opisany kinematyką postaci (2.68), ze zbiorem konfiguracji osobliwych (2.137). Określmy postacie normalne kinematyki podwójnego wahadła. Oczywiście, w otoczeniu konfiguracji regularnych kinematyka jest równoważna liniowej postaci normalnej

$$\mathbf{k}_0(\mathbf{x}) = (x_1, x_2)^T,$$

<sup>¶</sup>Tzn. takiej, że  $y_{dj}(t_s) = 0$ .



na mocy wniosku 3.2.1. Manipulator przyjmuje konfigurację osobliwą, gdy jego drugie ramię jest w pełni wyprostowane ( $x_2 \bmod 2\pi = 0$ ) lub całkowicie złożone ( $x_2 \bmod 2\pi = \pi$ ). Na podstawie analizy jacobianu analitycznego manipulatora (2.82) łatwo zauważyć, że wszystkie konfiguracje osobliwe kinematyki (2.68) są korzędu 1. Zmiana kolejności współrzędnych przegubowych zgodnie z zależnością

$$(\bar{x}_1, \bar{x}_2) = (x_2, x_1),$$

prowadzi do zdefiniowania kinematyki

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \mathbf{k}(\bar{\mathbf{x}}) = \begin{pmatrix} l_1 \cos \bar{x}_2 + l_2 \cos(\bar{x}_1 + \bar{x}_2) \\ l_1 \sin \bar{x}_2 + l_2 \sin(\bar{x}_1 + \bar{x}_2) \end{pmatrix}, \quad (3.92)$$

której jacobian analityczny

$$\mathbf{J}^a(\bar{\mathbf{x}}) = \begin{bmatrix} -l_2 \sin(\bar{x}_1 + \bar{x}_2) & -l_1 \sin \bar{x}_2 - l_2 \sin(\bar{x}_1 + \bar{x}_2) \\ l_2 \cos(\bar{x}_1 + \bar{x}_2) & l_1 \cos \bar{x}_2 + l_2 \cos(\bar{x}_1 + \bar{x}_2) \end{bmatrix},$$

a zbiór konfiguracji osobliwych  $S_1 = \{\bar{\mathbf{x}} = (\bar{x}_1, \bar{x}_2)^T \in \mathbb{R}^2 \mid \sin \bar{x}_1 = 0\}$ . Kinematyka (3.92) spełnia warunek (3.79), jeśli  $\sin \bar{x}_{02} \neq 0$ . Rozwiązanie równania  $\mathbf{J}^a(\bar{\mathbf{x}}_0)\mathbf{v} = 0$  dla  $\bar{\mathbf{x}}_0 = (\bar{x}_{01}, \bar{x}_{02})^T \in S_1$  można wybrać jako

$$\mathbf{v} = \begin{cases} (l_1 + l_2, -l_2)^T, & \text{dla } \bar{x}_{01} \bmod 2\pi = 0, \\ (l_1 - l_2, l_2)^T, & \text{dla } \bar{x}_{01} \bmod 2\pi = \pi. \end{cases}$$

Różniczka wyznacznika jacobianu analitycznego ma postać

$$d(\det \mathbf{J}^a)(\bar{\mathbf{x}}_0) = \begin{cases} (l_1 l_2, 0), & \text{dla } \bar{x}_{01} \bmod 2\pi = 0, \\ (-l_1 l_2, 0), & \text{dla } \bar{x}_{01} \bmod 2\pi = \pi. \end{cases}$$

W konsekwencji, lewa strona nierówności (3.82) jest równa

$$\begin{cases} l_1 l_2 (l_1 + l_2), & \text{dla } \bar{x}_{01} \bmod 2\pi = 0, \\ l_1 l_2 (l_2 - l_1), & \text{dla } \bar{x}_{01} \bmod 2\pi = \pi, \end{cases}$$

i dla  $\bar{x}_{01} \bmod 2\pi = 0$  jest zawsze różna od zera<sup>||</sup>, a dla  $\bar{x}_{01} \bmod 2\pi = \pi$  jest niezerowa wtedy i tylko wtedy, gdy  $l_1 \neq l_2$ . Przeprowadzenie podobnej analizy po dodatkowej zmianie kolejności współrzędnych zadaniowych kinematyki (wówczas warunek (3.79) jest spełniony, jeśli  $\cos \bar{x}_{02} \neq 0$ ) prowadzi na podstawie wniosku 3.2.2 do następującego stwierdzenia.

<sup>||</sup>Przyjmujemy, że  $l_1, l_2 > 0$ .

**Wniosek 3.2.4** *Kinematyka manipulatora typu podwójne wahadło posiada paraboliczną postać normalną*

$$\mathbf{k}_0(\mathbf{x}) = (x_1, x_2^2)^T,$$

w otoczeniu konfiguracji osobliwych  $x_{02} \bmod 2\pi = 0$ , a przy  $l_1 \neq l_2$  także w otoczeniu konfiguracji osobliwych  $x_{02} \bmod 2\pi = \pi$ .

Widzimy, że jedynymi konfiguracjami manipulatora, dla których nie określiliśmy dotychczas postaci normalnej są konfiguracje osobliwe  $x_{02} \bmod 2\pi = \pi$  przy  $l_1 = l_2$ . Przyjmijmy  $x_{02} = \pi$ , co przy kinematyce (3.92) odpowiada konfiguracji osobliwej  $\bar{\mathbf{x}}_0 = (\pi, \bar{x}_{02})^T$ . Do określenia postaci normalnej w tej sytuacji posłużymy się wnioskiem 3.2.3. Zgodnie z zależnością (3.84), zdefiniujmy w otoczeniu tej konfiguracji osobliwej układ współrzędnych

$$\xi = \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \boldsymbol{\varphi}(\bar{\mathbf{x}}) = \begin{pmatrix} l_1 \cos \bar{x}_2 + l_2 \cos(\bar{x}_1 + \bar{x}_2) \\ \bar{x}_2 \end{pmatrix}. \quad (3.93)$$

Odwzorowanie (3.93) jest wzajemnie jednoznaczne dla otoczenia punktu osobliwego  $\bar{\mathbf{x}}_0 = (\bar{x}_{01}, \bar{x}_{02})^T$  postaci

$$\mathbf{U}_\alpha = \left(\frac{\pi}{2} + \alpha, \frac{3\pi}{2} - \alpha\right) \times \left(\frac{\pi}{2} - \alpha, \frac{\pi}{2} + \alpha\right)$$

przy  $0 < \alpha < \pi/2$ . Dla takiego otoczenia  $\mathbf{U}_\alpha \cap S_1 = \{\pi\} \times \left(\frac{\pi}{2} - \alpha, \frac{\pi}{2} + \alpha\right)$ , a zatem

$$\begin{aligned} \boldsymbol{\varphi}(\mathbf{U}_\alpha \cap S_1) &= \{(l_1 \cos \bar{x}_2 + l_2 \cos(\bar{x}_1 + \bar{x}_2), \bar{x}_2)^T \mid \bar{x}_1 = \pi, \frac{\pi}{2} - \alpha < \\ &< \bar{x}_2 < \frac{\pi}{2} + \alpha\} = \{((l_1 - l_2) \cos \xi_2, \xi_2)^T \mid \frac{\pi}{2} - \alpha < \xi_2 < \frac{\pi}{2} + \alpha\}. \end{aligned}$$

Z drugiej strony, otoczenie  $\mathbf{V}_\alpha$  punktu  $\xi_0 = \boldsymbol{\varphi}(\bar{\mathbf{x}}_0) = ((l_1 - l_2) \cos \bar{x}_{02}, \bar{x}_{02})^T$  przyjmuje postać

$$\begin{aligned} \mathbf{V}_\alpha = \boldsymbol{\varphi}(\mathbf{U}_\alpha) &= \{(\xi_1, \xi_2)^T \mid \xi_1 = l_1 \cos \bar{x}_2 + l_2 \cos(\bar{x}_1 + \bar{x}_2), \\ &\xi_2 = \bar{x}_2, (\bar{x}_1, \bar{x}_2)^T \in \mathbf{U}_\alpha\}, \end{aligned}$$

skąd otrzymujemy

$$\mathbf{V}_{\alpha 1} = \{((l_1 - l_2) \cos \bar{x}_{02}, \xi_2)^T \mid \frac{\pi}{2} - \alpha < \xi_2 < \frac{\pi}{2} + \alpha\}.$$

Widać więc, że dla  $l_1 = l_2$  jest prawdziwa zależność  $\boldsymbol{\varphi}(\mathbf{U}_\alpha \cap S_1) = \mathbf{V}_{\alpha 1}$ , co oznacza spełnienie warunku (3.86).

Dla długości ramion  $l_1 = l_2$  różniczka wyznacznika jacobianu analitycznego w punkcie  $\bar{\mathbf{x}}_0 = (\pi, \bar{x}_{02})^\top$  ma postać

$$d(\det \mathbf{J}^a)(\bar{\mathbf{x}}_0) = (-l_1^2, 0),$$

a macierz

$$\left( \frac{\partial \boldsymbol{\varphi}}{\partial \bar{\mathbf{x}}}(\bar{\mathbf{x}}_0) \right)^{-1} = \begin{bmatrix} \frac{1}{l_1 \sin x_{02}} & 0 \\ 0 & 1 \end{bmatrix}.$$

Widać stąd, że warunek (3.87) jest spełniony, gdyż

$$d(\det \mathbf{J}^a)(\bar{\mathbf{x}}_0) \left( \frac{\partial \boldsymbol{\varphi}}{\partial \bar{\mathbf{x}}}(\bar{\mathbf{x}}_0) \right)^{-1}_{\text{kol}} = \frac{-l_1}{\sin \bar{x}_{02}} \neq 0.$$

W ten sposób ustaliliśmy postać normalną kinematyki manipulatora typu podwójne wahadło dla konfiguracji osobliwej  $x_{02} = \pi$ , przy  $l_1 = l_2$  i  $\sin x_{01} \neq 0$ . Po przeprowadzeniu podobnej analizy, przy dodatkowej zamianie miejscami współrzędnych zadaniowych  $y_1$  i  $y_2$  kinematyki, dochodzimy do następującej konkluzji.

**Wniosek 3.2.5** *W otoczeniu konfiguracji osobliwych  $x_{02} \bmod 2\pi = \pi$  kinematyka manipulatora typu podwójne wahadło, którego ramiona mają jednakowe długości ( $l_1 = l_2$ ), jest równoważna hiperbolicznej postaci normalnej*

$$\mathbf{k}_0(\mathbf{x}) = (x_1, x_1 x_2)^\top. \quad \blacksquare$$

### Przykład 3.2.3 (Manipulator typu podwójne wahadło)

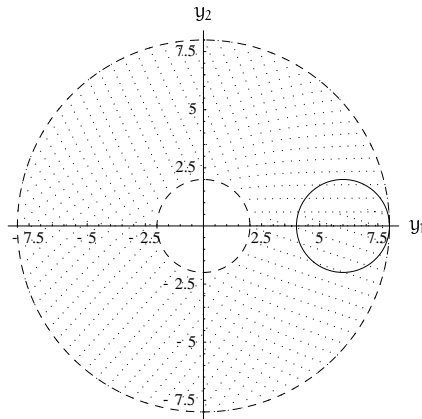
W tym przykładzie opiszemy pierwszy, przybliżony sposób rozwiązania odwrotnego zadania kinematyki z wykorzystaniem metody postaci normalnych. Drugi sposób, w którym korzystamy z rozwiązania regularnego odwrotnego zadania kinematyki w postaci jawnej (przykład 3.1.1) i wniosków płynących z metody postaci normalnych, przedstawia przykład następny. W obliczeniach przyjmujemy  $l_1 = 5$ ,  $l_2 = 3$ .

Niech zadaniem manipulatora będzie przemieszczenie efektora wzdłuż trajektorii zadaniowej mającej postać okręgu

$$\mathbf{y}_d(t) = \begin{pmatrix} y_{d1}(t) \\ y_{d2}(t) \end{pmatrix} = \begin{pmatrix} 6 + 2 \sin(\frac{\pi}{5}t) \\ 2 \cos(\frac{\pi}{5}t) \end{pmatrix}, \quad (3.94)$$

w przedziale czasu  $\mathcal{I} = [0, 10]$ \*\*.

\*\*Porównaj z trajektorią z przykładu 3.1.5.



Rysunek 3.24 Ścieżka zadana w przestrzeni roboczej manipulatora typu podwójne wahadło.

strzeni roboczej manipulatora (obszar zakropkowany) odpowiadającą zadanej trajektorii. Naszym celem jest znalezienie trajektorii przegubowej manipulatora pozwalającej na realizację trajektorii (3.94). Zauważmy, że trajektoria ta została tak dobrana, aby jej realizacja wymagała wprowadzenia manipulatora w konfigurację osobliwą, co powoduje, że próba rozwiązania zadania metodą jacobianu odwrotnego kończy się niepowodzeniem.

Zgodnie z pierwszym krokiem algorytmu metody postaci normalnych, musimy wyznaczyć zbiór chwil, w których trajektoria (3.94) przecina obraz zbioru konfiguracji osobliwych  $\mathbf{k}(S)$  przy  $S$  zdefiniowanym wzorem (2.137). Łatwo pokazać, że dla manipulatora typu podwójne wahadło dzieje się tak wtedy i tylko wtedy, gdy

$$y_{d1}^2(t) + y_{d2}^2(t) = (l_1 \pm l_2)^2,$$

co w analizowanym przykładzie sprowadza się do jednego z dwóch warunków:  $40 + 24 \sin(\pi/5t) = 64$ , albo  $40 + 24 \sin(\pi/5t) = 4$ . Pierwszy warunek jest spełniony dla  $t_{s1} = 2.5$ , gdy manipulator osiąga konfigurację  $\mathbf{x}_s = (0, 0)^T$ . Zbliżanie się trajektorii przegubowej do konfiguracji osobliwej potwierdza próba rozwiązania zadania odwrotnego metodą jacobianu odwrotnego; podczas numerycznego rozwiązywania układu równań (3.20) otrzymujemy komunikat o złym uwarunkowaniu układu spowodowanym wystąpieniem osobliwości, co uniemożliwia jego rozwiązanie (w przeprowadzonych symulacjach komunikat taki otrzymaliśmy w chwili  $t = 2.496$ ).

Zajmiemy się obecnie wyznaczeniem osobliwego fragmentu poszukiwanej trajektorii, zgodnie z algorytmem metody postaci normalnych. Krok 3

algorytmu wymaga wyznaczenia postaci normalnej kinematyki. Na podstawie wniosku 3.2.4 wiemy, że w konfiguracji osobliwej  $\mathbf{x}_s = (0, 0)^T$  kinematyka badanego manipulatora jest równoważna parabolicznej postaci normalnej. Naszym zadaniem jest zatem znalezienie układów współrzędnych  $\xi = \varphi(\mathbf{x})$ ,  $\eta = \psi(\mathbf{y})$ , w których będzie spełniona zależność (3.76). Odwzorowań  $\varphi$ ,  $\psi$  będziemy szukać w postaci rozwinięć w szereg Taylora. Całą operację przeprowadzimy w dwóch etapach. W pierwszym etapie znajdziemy układy współrzędnych  $\bar{\mathbf{x}} = \bar{\varphi}(\mathbf{x})$ ,  $\bar{\mathbf{y}} = \bar{\psi}(\mathbf{y})$ , które sprowadzają rozpatrywaną kinematykę do postaci  $\bar{\mathbf{k}}(\bar{\mathbf{x}}) = (\bar{x}_1, r(\bar{\mathbf{x}}))^T$ <sup>††</sup>, w drugim — układy współrzędnych  $\xi = \hat{\varphi}(\bar{\mathbf{x}})$ ,  $\eta = \hat{\psi}(\bar{\mathbf{y}})$  redukujące kinematykę  $\bar{\mathbf{k}}(\bar{\mathbf{x}})$  do parabolicznej postaci normalnej  $\mathbf{k}_0(\xi) = (\xi_1, \xi_2^2)^T$ . Odwzorowania  $\varphi(\mathbf{x})$ ,  $\psi(\mathbf{y})$  znajdziemy jako złożenia

$$\begin{cases} \xi = \varphi(\mathbf{x}) = \hat{\varphi} \circ \bar{\varphi}(\mathbf{x}) \\ \eta = \psi(\mathbf{y}) = \hat{\psi} \circ \bar{\psi}(\mathbf{y}). \end{cases} \quad (3.95)$$

Po zmianie kolejności współrzędnych zadaniowych kinematyki (2.68) mającej na celu spełnienie warunku (3.79), kinematyka manipulatora przyjmuje postać

$$\mathbf{y} = \mathbf{k}(\mathbf{x}) = \begin{pmatrix} 5s_1 + 3s_{12} \\ 5c_1 + 3c_{12} \end{pmatrix}. \quad (3.96)$$

Po rozwinięciu kinematyki (3.96) w szereg Taylora w otoczeniu punktu osobliwego  $\mathbf{x}_s = (x_{s1}, x_{s2})^T = (0, 0)^T$  otrzymujemy

$$\begin{aligned} \mathbf{y} = \mathbf{k}(\mathbf{x}) &= \begin{pmatrix} k_1(\mathbf{x}) \\ k_2(\mathbf{x}) \end{pmatrix} = \\ &= \begin{pmatrix} 8x_1 - \frac{4}{3}x_1^3 + 3x_2 - \frac{3}{2}x_1^2x_2 - \frac{3}{2}x_1x_2^2 - \frac{1}{2}x_2^3 + o^4(\mathbf{x}) \\ 8 - 4x_1^2 - 3x_1x_2 - \frac{3}{2}x_2^2 + o^4(\mathbf{x}) \end{pmatrix}, \end{aligned}$$

gdzie  $o^n(\mathbf{x})$  oznacza małą funkcję rzędu  $n$ -tego ze względu na składowe wektora  $\mathbf{x}$ <sup>‡‡</sup>. Wybór odwzorowań  $\bar{\varphi}(\mathbf{x})$  i  $\bar{\psi}(\mathbf{y})$  w postaci

$$\begin{cases} \bar{\mathbf{x}} = \bar{\varphi}(\mathbf{x}) = (k_1(\mathbf{x} - \mathbf{x}_s), x_2 - x_{s2})^T \\ \bar{\mathbf{y}} = \bar{\psi}(\mathbf{y}) = \mathbf{y} - \mathbf{k}(\mathbf{x}_s), \end{cases}$$

<sup>††</sup>W ogólności, dla kinematyki nieredundantnej o liczbie stopni swobody  $n$ , do postaci  $\bar{\mathbf{k}}(\bar{\mathbf{x}}) = (\bar{x}_1, \dots, \bar{x}_{n-1}, r(\bar{\mathbf{x}}))^T$ , zwanej *prenormalną*.

<sup>‡‡</sup>Tzn. w rozwinięciu Taylora  $o^n(\mathbf{x})$  najniższa potęga  $\mathbf{x}$  wynosi  $n$ .

czyli

$$\begin{cases} \bar{\mathbf{x}} = \begin{pmatrix} 8x_1 - \frac{4}{3}x_1^3 + 3x_2 - \frac{3}{2}x_1^2x_2 - \frac{3}{2}x_1x_2^2 - \frac{1}{2}x_2^3 + o^4(\mathbf{x}) \\ x_2 \end{pmatrix} \\ \bar{\mathbf{y}} = \begin{pmatrix} y_1 \\ y_2 - 8 \end{pmatrix}, \end{cases}$$

prowadzi do uzyskania kinematyki  $\bar{\mathbf{k}}(\bar{\mathbf{x}})$  opisanej wyrażeniem

$$\bar{\mathbf{y}} = \bar{\mathbf{k}}(\bar{\mathbf{x}}) = \bar{\boldsymbol{\psi}} \circ \mathbf{k} \circ \bar{\boldsymbol{\varphi}}^{-1}(\bar{\mathbf{x}}) = \begin{pmatrix} \bar{x}_1 \\ -\frac{1}{16}\bar{x}_1^2 - \frac{15}{16}\bar{x}_2^2 + o^4(\bar{\mathbf{x}}) \end{pmatrix}, \quad (3.97)$$

gdzie

$$\bar{\boldsymbol{\varphi}}^{-1}(\bar{\mathbf{x}}) = \begin{pmatrix} \frac{1}{8}\bar{x}_1 + \frac{1}{3072}\bar{x}_1^3 - \frac{3}{8}\bar{x}_2 + \frac{15}{1024}\bar{x}_1\bar{x}_2^2 + \frac{5}{512}\bar{x}_2^3 + o^4(\bar{\mathbf{x}}) \\ \bar{x}_2 \end{pmatrix}.$$

Będziemy teraz szukać pary odwzorowań  $\boldsymbol{\xi} = \hat{\boldsymbol{\varphi}}(\bar{\mathbf{x}}) = (\hat{\varphi}_1(\bar{\mathbf{x}}), \hat{\varphi}_2(\bar{\mathbf{x}}))^T$  i  $\boldsymbol{\eta} = \hat{\boldsymbol{\psi}}(\bar{\mathbf{y}}) = (\hat{\psi}_1(\bar{\mathbf{y}}), \hat{\psi}_2(\bar{\mathbf{y}}))^T$  spełniających równanie (3.76) dla  $\mathbf{k}(\mathbf{x}) = \bar{\mathbf{k}}(\mathbf{x})$  i parabolicznej postaci normalnej  $\mathbf{k}_0(\boldsymbol{\xi}) = (\xi_1, \xi_2^2)^T$ . Na wstępie zauważmy, że dla pierwszych współrzędnych poszukiwanych przekształceń mamy  $\xi_1 = \hat{\varphi}_1(\bar{\mathbf{x}}) = \bar{x}_1$ ,  $\eta_1 = \hat{\psi}_1(\bar{\mathbf{y}}) = \bar{y}_1$ . Aby znaleźć  $\hat{\varphi}_2(\bar{\mathbf{x}})$  i  $\hat{\psi}_2(\bar{\mathbf{y}})$ , założmy że rozwinięcie w szereg Taylora odwzorowania  $\hat{\boldsymbol{\varphi}}(\bar{\mathbf{x}})$  ma postać

$$\boldsymbol{\xi} = \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \hat{\boldsymbol{\varphi}}(\bar{\mathbf{x}}) = \begin{pmatrix} \bar{x}_1 \\ \hat{\varphi}_2(\bar{\mathbf{x}}) \end{pmatrix},$$

gdzie  $\hat{\varphi}_2(\bar{\mathbf{x}}) = a_1\bar{x}_1 + a_2\bar{x}_2 + a_3\bar{x}_1^2 + a_4\bar{x}_1\bar{x}_2 + a_5\bar{x}_2^2 + o^3(\bar{\mathbf{x}})$ ,  $a_1 \div a_5$  — parametry rozwinięcia. Wówczas, złożenie kinematyki  $\mathbf{k}_0(\boldsymbol{\xi})$  z  $\hat{\boldsymbol{\varphi}}(\bar{\mathbf{x}})$  prowadzi do wyrażenia

$$\boldsymbol{\eta} = \begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix} = \mathbf{k}_0 \circ \hat{\boldsymbol{\varphi}}(\bar{\mathbf{x}}) = \begin{pmatrix} \bar{x}_1 \\ \hat{\varphi}_2^2(\bar{\mathbf{x}}) \end{pmatrix}, \quad (3.98)$$

gdzie  $\hat{\varphi}_2^2(\bar{\mathbf{x}}) = a_1^2\bar{x}_1^2 + 2a_1a_2\bar{x}_1\bar{x}_2 + a_2^2\bar{x}_2^2 + 2a_1a_3\bar{x}_1^3 + 2(a_2a_3 + a_1a_4)\bar{x}_1^2\bar{x}_2 + 2(a_2a_4 + a_1a_5)\bar{x}_1\bar{x}_2^2 + 2a_2a_5\bar{x}_2^3 + o^4(\bar{\mathbf{x}})$ . Z drugiej strony, rozwinięcie w szereg Taylora odwzorowania  $\hat{\boldsymbol{\psi}}(\bar{\mathbf{y}})$  postaci

$$\boldsymbol{\eta} = \begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix} = \hat{\boldsymbol{\psi}}(\bar{\mathbf{y}}) = \begin{pmatrix} \bar{y}_1 \\ \hat{\psi}_2(\bar{\mathbf{y}}) \end{pmatrix},$$

gdzie  $\hat{\psi}_2(\bar{\mathbf{y}}) = b_1\bar{y}_2 + b_2\bar{y}_1 + b_3\bar{y}_1^2 + b_4\bar{y}_1^3 + o^4(\bar{y}_1)$  zawiera współczynnik  $b_1 = \pm 1$  i nieznane parametry  $b_2$ – $b_4$ <sup>†</sup>, po złożeniu z  $\bar{\mathbf{k}}(\bar{\mathbf{x}})$  daje

$$\boldsymbol{\eta} = \begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix} = \hat{\boldsymbol{\psi}} \circ \bar{\mathbf{k}}(\bar{\mathbf{x}}) = \begin{pmatrix} \bar{x}_1 \\ \psi_2(\bar{\mathbf{k}}(\bar{\mathbf{x}})) \end{pmatrix}, \quad (3.99)$$

gdzie  $\psi_2(\bar{\mathbf{k}}(\bar{\mathbf{x}})) = b_2\bar{x}_1 - \frac{1}{16}b_1\bar{x}_1^2 + b_3\bar{x}_1^2 + b_4\bar{x}_1^3 - \frac{15}{16}b_1\bar{x}_2^2 + o^4(\bar{\mathbf{x}})$ . Porównanie współczynników przy odpowiednich potęgach w (3.98) i (3.99) pozwala stwierdzić, że kinematyka (3.97) przyjmuje paraboliczną postać normalną we współrzędnych

$$\begin{cases} \boldsymbol{\xi} = \hat{\boldsymbol{\varphi}}(\bar{\mathbf{x}}) = \begin{pmatrix} \bar{x}_1 \\ \frac{\sqrt{15}}{4}\bar{x}_2 + o^3(\bar{\mathbf{x}}) \end{pmatrix} \\ \boldsymbol{\eta} = \hat{\boldsymbol{\psi}}(\bar{\mathbf{y}}) = \begin{pmatrix} \bar{y}_1 \\ -\bar{y}_2 - \frac{1}{16}\bar{y}_1^2 + o^4(\bar{y}_1) \end{pmatrix}. \end{cases}$$

Analogicznie, na podstawie zależności (3.95), możemy stwierdzić to samo o kinematyce (2.68) wyrażonej we współrzędnych

$$\begin{cases} \boldsymbol{\xi} = \boldsymbol{\varphi}(\mathbf{x}) = \begin{pmatrix} 8x_1 + 3x_2 + o^3(\mathbf{x}) \\ \frac{\sqrt{15}}{4}x_2 + o^3(\mathbf{x}) \end{pmatrix} \\ \boldsymbol{\eta} = \boldsymbol{\psi}(\mathbf{y}) = \begin{pmatrix} y_2 \\ 8 - y_1 - \frac{1}{16}y_2^2 + o^4(y_2) \end{pmatrix}. \end{cases}$$

Jak wynika z kroku 6 algorytmu metody postaci normalnych, do rozwiązania zadania odwrotnego będziemy potrzebować odwzorowania  $\boldsymbol{\varphi}^{-1}$ , które ma postać

$$\mathbf{x} = \boldsymbol{\varphi}^{-1}(\boldsymbol{\xi}) = \begin{pmatrix} \frac{1}{8}\xi_1 - \frac{\sqrt{3}}{\sqrt{20}}\xi_2 + o^3(\boldsymbol{\xi}) \\ \frac{4}{\sqrt{15}}\xi_2 + o^3(\boldsymbol{\xi}) \end{pmatrix}. \quad (3.100)$$

Wyliczone przekształcenia umożliwiają rozwiązanie zadania w otoczeniu konfiguracji osobliwych. Analiza przebiegu trajektorii zadanej pozwala przyjąć, że cała trajektoria leży w tym otoczeniu, dzięki czemu rozwiązanie zadania będzie się składać jedynie z części osobliwej. Zgodnie z krokiem 4 algorytmu metody postaci normalnych, nałożenie odwzorowania  $\boldsymbol{\psi}(\mathbf{y})$  danego

<sup>†</sup>Szczegółowa analiza dyfeomorfizmu  $\hat{\boldsymbol{\psi}}(\bar{\mathbf{y}})$  prowadzi do wniosku, że dla dowolnego  $n$  jest on dany jako  $\hat{\boldsymbol{\psi}}(\bar{\mathbf{y}}) = (\bar{y}_1, \dots, \bar{y}_{n-1}, \pm\bar{y}_n + f(\bar{y}_1, \dots, \bar{y}_{n-1}))^T$ , gdzie  $f(\bar{y}_1, \dots, \bar{y}_{n-1})$  jest pewną funkcją analityczną, taką że  $f(\mathbf{0}) = 0$ .

przez (3.2.3) na trajektorię zadaną (3.94) pozwala na wyliczenie trajektorii zadaniowej  $\eta_d(t)$  dla parabolicznej postaci normalnej. Rozwiązanie osobliwego odwrotnego zadania kinematyki dla tej postaci normalnej według zależności (3.90) (krok 5 algorytmu), a następnie wyliczenie odwzorowania (3.100) wzdłuż otrzymanego rozwiązania (krok 6), prowadzi do trajektorii pokazanych na rysunku 3.25. Przedstawione rozwiązanie uzyskano przy rozwinięciu odwzorowań (3.2.3) w szereg Taylora z dokładnością do wyrazów rzędu siódmego. W wyrażeniu (3.90) przyjęto znak plus. Błędy położenia i prędkości efektora manipulatora przy tak uzyskanej trajektorii ilustruje rysunek 3.26. Rysunek 3.27 pokazuje ścieżkę rzeczywistą (linia ciągła) w przestrzeni roboczej uzyskaną w wyniku rozwiązania zadania, przedstawioną na tle ścieżki zadanej (linia przerywana). Moduł błędu położenia efektora osiąga maksymalną wartość 0.122 dla chwili  $t = 7.88$ . Dodajmy, że dla czasów  $t \in [1, 4]$  maksymalna wartość błędu spada do poziomu  $4 \times 10^{-5}$  (dla czasów  $t \in [2, 3]$ <sup>‡</sup> jest to już tylko  $2 \times 10^{-9}$ , to znaczy dużo poniżej poziomu błędów numerycznych towarzyszących metodom jakobianowym). W celu zmniejszenia błędów należałoby wykorzystać jedynie fragment przedstawionego rozwiązania osobliwego, a pozostałą część trajektorii znaleźć zgodnie z krokiem 2 algorytmu metody postaci normalnych lub określić dyfeomorfizmy  $\varphi$ ,  $\psi$  z większą dokładnością. Można też skorzystać z metody postaci normalnych w sposób zilustrowany następnym przykładem. ■

### Przykład 3.2.4 (Manipulator typu podwójne wahadło)

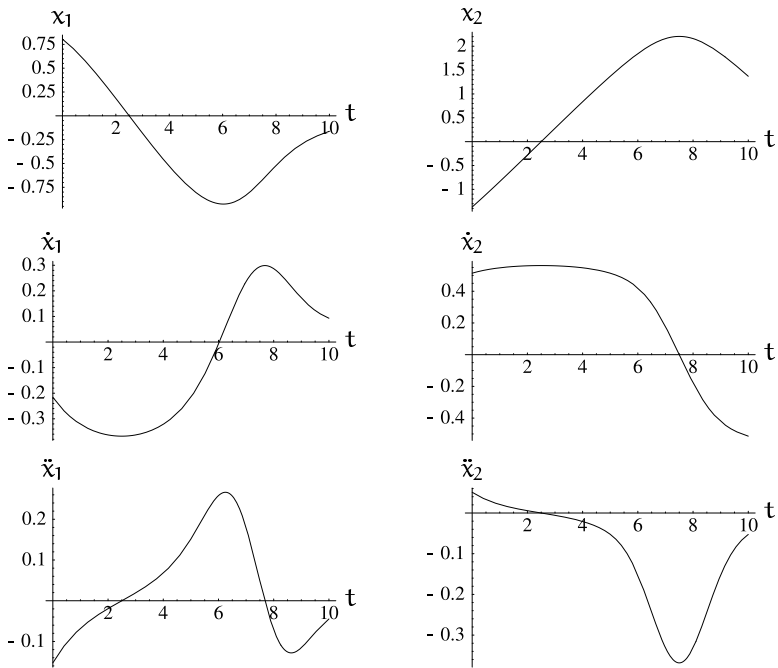
W tym przykładzie przedstawimy rozwiązanie osobliwego odwrotnego zadania kinematyki otrzymane metodą postaci normalnych połączoną z bezpośrednią metodą algebraiczną<sup>§</sup>.

Jak wynika z wyrażen (3.89), (3.90) opisujących rozwiązania osobliwego odwrotnego zadania kinematyki dla parabolicznej postaci normalnej, w otoczeniu konfiguracji osobliwych tego typu możliwe są cztery różne trajektorie przegubowe realizujące zadany ruch. Krótka analiza rozwiązania odwrotnego zadania kinematyki dla manipulatora typu podwójne wahadło przy użyciu bezpośredniego podejścia algebraicznego, opisanego za pomocą wzoru (3.10), prowadzi do wniosku, że w otoczeniu konfiguracji osobliwych rozwiązanie to opisuje jedynie dwie trajektorie realizujące zadany ruch. Są to trajektorie typu (3.89). Aby uzyskać w postaci jawnej dwa pozostałe rozwiązania (typu (3.90)), należy zmodyfikować postać funkcji  $s_2$

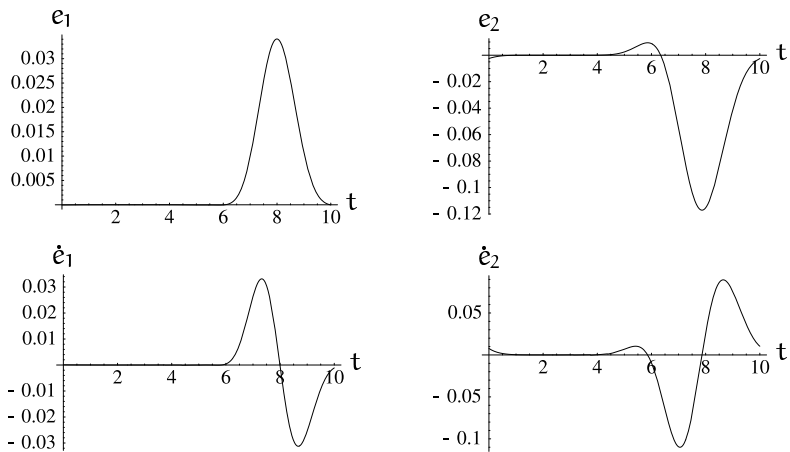
<sup>‡</sup>Więc wciąż w dużej odległości od konfiguracji osobliwej.

<sup>§</sup>Zobacz przykład 3.1.1.

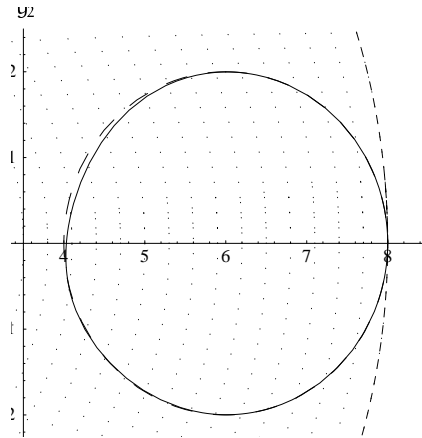




**Rysunek 3.25** Przybliżone rozwiązanie osobliwego zadania odwrotnego dla podwójnego wahadła uzyskane metodą postaci normalnych.



**Rysunek 3.26** Błędy położenia i prędkości efektora przy metodzie postaci normalnych w wersji przybliżonej.



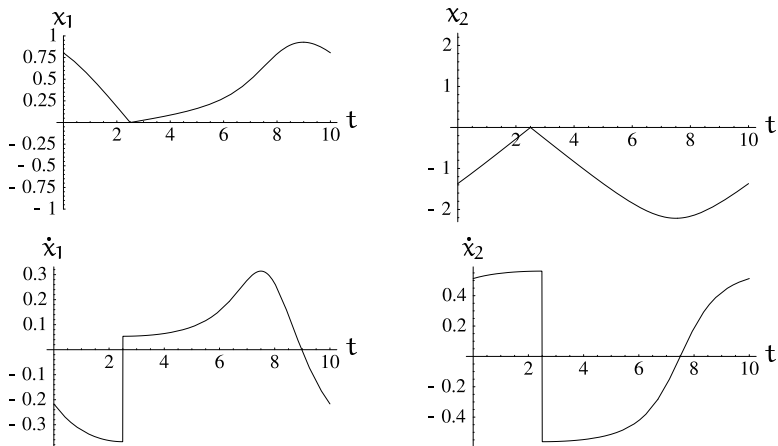
**Rysunek 3.27** Ścieżka zadana i ścieżka rzeczywista efektora manipulatora typu podwójne wahadło.

występującej w równaniach (3.10) definiując ją jako

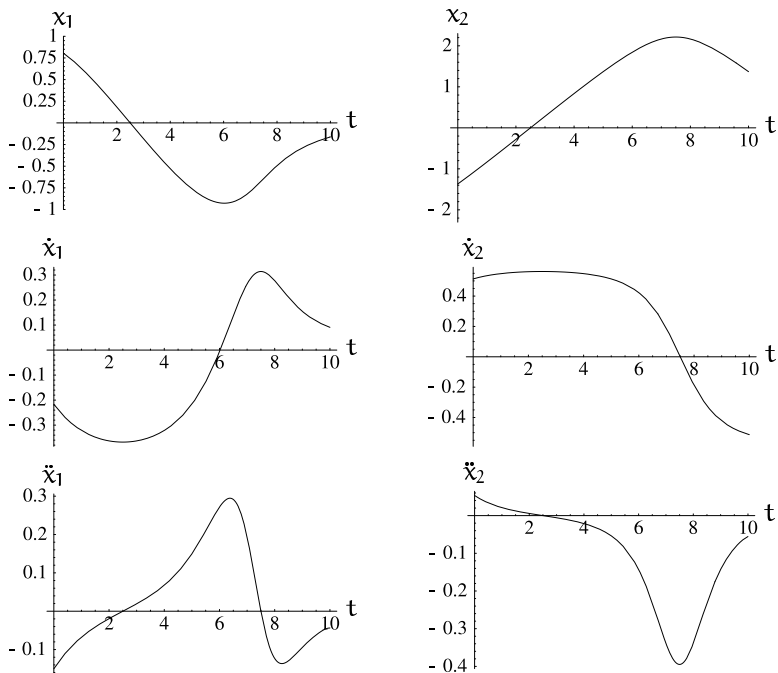
$$s_2 = \pm \prod_{t_{si} \in T} \operatorname{sgn}(t - t_{si}) \sqrt{1 - c_2^2},$$

gdzie  $T$  jest zbiorem chwil opisanym w kroku 1 algorytmu na stronie 142. Jeśli trajektoria zadaniowa wymaga jednokrotnego wprowadzenia manipulatora w konfigurację osobliwą, wówczas funkcja  $s_2 = \pm \operatorname{sgn}(t - t_s) \sqrt{1 - c_2^2}$ , gdzie  $t_s$  oznacza chwilę osiągnięcia przez manipulator konfiguracji osobliwej.

Załóżmy, że zadaniem manipulatora jest ponownie realizacja trajektorii zadanej wyrażeniem (3.94). Rozwiązanie tego zadania bezpośrednio metodą algebraiczną (zależność (3.10) z funkcją  $s_2$  daną przez (3.7)) przedstawiliśmy na rysunku 3.28. Rozwiązanie uzyskane po uwzględnieniu omówionych wyżej modyfikacji przedstawiono na rysunku 3.29. Porównanie tych rozwiązań wykazuje, że, w przeciwieństwie do pierwszego, drugie z nich jest gładkie w całej swojej dziedzinie. W chwili  $t_s = 2.5$ , gdy manipulator osiąga konfigurację osobliwą, pojawia się skok prędkości trajektorii przegubowej (któremu teoretycznie towarzyszą nieskończone przyspieszenia w przegubach), powodujący szarpnięcie przegubów manipulatora. Na zakończenie dodajmy, że obu przedstawionym rozwiązaniom towarzyszyły błędy położenia efektora na poziomie dokładności prowadzonych obliczeń ( $10^{-16}$ ). ■



**Rysunek 3.28** Rozwiązanie osobliwego zadania odwrotnego dla podwójnego wahadła uzyskane bezpośrednio metodą algebraiczną.



**Rysunek 3.29** Dokładne rozwiązanie osobliwego zadania odwrotnego dla podwójnego wahadła uzyskane metodą postaci normalnych.

**Przykład 3.2.5 (Manipulator IRb-6 na torze jezdny)**

Rozpatrzmy manipulator IRb-6 zamontowany na torze jezdny, opisany kinematyką (2.76), którego zbiór konfiguracji osobliwych jest zdefiniowany przez (2.144). Po zmianie kolejności współrzędnych

$$\begin{cases} (\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4, \bar{x}_5, \bar{x}_6) = (x_6, x_2, x_3, x_4, x_5, x_1) \\ (\bar{y}_1, \bar{y}_2, \bar{y}_3, \bar{y}_4, \bar{y}_5, \bar{y}_6) = (y_6, y_2, y_3, y_4, y_5, y_1) \end{cases}$$

otrzymujemy kinematykę manipulatora w postaci

$$\bar{\mathbf{y}} = \bar{\mathbf{k}}(\bar{\mathbf{x}}) = \begin{pmatrix} \pi - \bar{x}_1 \\ -d_1 + d_6 c_5 - a_2 s_3 - a_3 s_4 \\ \bar{x}_6 + s_2(a_2 c_3 + a_3 c_4 + d_6 s_5) \\ -\bar{x}_2 \\ \pi - \bar{x}_5 \\ c_2(a_2 c_3 + a_3 c_4 + d_6 s_5) \end{pmatrix},$$

gdzie  $s_i, c_i$  oznacza odpowiednio  $\sin \bar{x}_i$  i  $\cos \bar{x}_i$ . Zauważmy, że dokonana zmiana współrzędnych nie zmienia zbioru konfiguracji osobliwych (2.144), złożonego jedynie z osobliwości korzędu 1. Dla tak określonej kinematyki lewa strona warunku (3.79) przyjmuje wartość  $a_2 a_3 \sin \bar{x}_2 \sin(\bar{x}_3 - \bar{x}_4)$ , która, po uwzględnieniu ograniczeń konstrukcyjnych manipulatora, w konfiguracjach osobliwych jest zawsze różna od zera. W otoczeniu

$$\mathbf{U} = \{(\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4, \bar{x}_5, \bar{x}_6)^T \mid -\pi \leq \bar{x}_1 \leq \pi, 0 < \bar{x}_2 \leq \frac{17}{18}\pi, \\ \frac{5}{18}\pi \leq \bar{x}_3 \leq \frac{13}{18}\pi, -\frac{2}{9}\pi \leq \bar{x}_4 \leq \frac{5}{36}, \frac{5}{18}\pi \leq \bar{x}_3 - \bar{x}_4 \leq \frac{13}{18}\pi, \\ 0 \leq \bar{x}_5 \leq \frac{1}{2}\pi, 0 \leq \bar{x}_6 \leq 1.5\}$$

punktu osobliwego  $\bar{\mathbf{x}}_0 = (\bar{x}_1, \pi/2, \bar{x}_3, \bar{x}_4, \bar{x}_5, \bar{x}_6)^T$  zdefiniujemy, zgodnie z równaniem (3.84), lokalny układ współrzędnych

$$\boldsymbol{\xi} = \begin{pmatrix} \xi_1 \\ \vdots \\ \xi_6 \end{pmatrix} = \boldsymbol{\varphi}(\bar{\mathbf{x}}) = \begin{pmatrix} \pi - \bar{x}_1 \\ -d_1 + d_6 c_5 - a_2 s_3 - a_3 s_4 \\ \bar{x}_6 + s_2(a_2 c_3 + a_3 c_4 + d_6 s_5) \\ -\bar{x}_2 \\ \pi - \bar{x}_5 \\ \bar{x}_6 \end{pmatrix}.$$

W nowych współrzędnych  $\boldsymbol{\xi}_0 = \boldsymbol{\varphi}(\bar{\mathbf{x}}_0) = (\pi - \bar{x}_{01}, -d_1 + d_6 c_{05} - a_2 s_{03} - a_3 s_{04}, \bar{x}_{06} + a_2 c_{03} + a_3 c_{04} + d_6 s_{05}, -\pi/2, \pi - \bar{x}_{05}, \bar{x}_{06})^T$ , gdzie  $s_{0i}, c_{0i}$  oznacza

$\sin \bar{x}_{0i}$  i  $\cos \bar{x}_{0i}$ . Otoczenie  $\mathbf{V} = \boldsymbol{\varphi}(\mathbf{U})$  punktu  $\boldsymbol{\xi}_0$  jest równe

$$\mathbf{V} = \left\{ (\pi - \bar{x}_1, -d_1 + d_6 c_5 - a_2 s_3 - a_3 s_4, \bar{x}_6 + s_2(a_2 c_3 + a_3 c_4 + d_6 s_5), \right. \\ \left. -x_2, \pi - \bar{x}_5, \bar{x}_6 \right)^T \mid -\pi \leq \bar{x}_1 \leq \pi, 0 < \bar{x}_2 \leq \frac{17}{18}\pi, \frac{5}{18}\pi \leq \bar{x}_3 \leq \frac{13}{18}\pi, \\ \left. -\frac{2}{9}\pi \leq \bar{x}_4 \leq \frac{5}{36}, \frac{5}{18}\pi \leq \bar{x}_3 - \bar{x}_4 \leq \frac{13}{18}\pi, 0 \leq \bar{x}_5 \leq \frac{1}{2}\pi, 0 \leq \bar{x}_6 \leq 1.5 \right\}.$$

Wybierając zbiór  $\mathbf{V}_4 = \mathbf{V}_{43} \times \{-\pi/2\} \times \mathbf{V}_{42}$ <sup>¶</sup> otrzymujemy  $\boldsymbol{\varphi}^{-1}(\mathbf{V}_4) = \mathbf{V}_{41} \times \{\pi/2\} \times \mathbf{V}_{44}$  i stwierdzamy, że  $\boldsymbol{\varphi}^{-1}(\mathbf{V}_4) \subset \mathbf{S}_1 \cap \mathbf{U}$  co oznacza, że jest spełniony warunek (3.86) z wniosku 3.2.3. Różniczka wyznacznika jacobianu analitycznego (2.143) wynosi

$$d(\det \mathbf{J}^a)(\bar{\mathbf{x}}_0) = (0, a_2 a_3 \sin(x_{03} - x_{04}), 0, 0, 0, 0).$$

Ponieważ element  $(\frac{\partial \boldsymbol{\varphi}}{\partial \mathbf{x}}(\mathbf{x}_0))_{24}^{-1}$  jest równy  $-1$ , warunek (3.87) jest także spełniony. W rezultacie, z wniosku 3.2.3 wynika następująca konkluzja.

**Wniosek 3.2.6** *W otoczeniu konfiguracji osobliwych kinematyka manipulatora IRb-6 zamontowanego na torze jezdny jest równoważna hiperbolicznej postaci normalnej*

$$\mathbf{k}_0(\mathbf{x}) = (x_1, \dots, x_5, x_6 x_4)^T. \quad (3.101)$$

■

### Przykład 3.2.6 (Manipulator IRb-6 na torze jezdny)

Załóżmy, że zadaniem manipulatora IRb-6 zamontowanego na torze jezdny jest przemieszczenie efektoru wzdłuż trajektorii zadaniowej, określonej w aspekcie współrzędnych  $x_0 y_0 z_0 \varphi \theta \psi$ , gdzie  $\varphi$ ,  $\theta$  i  $\psi$  są kątami Eulera Y-Z-Y, wyrażeniem

$$\mathbf{y}_d(t) = \begin{pmatrix} y_{d1}(t) \\ \vdots \\ y_{d6}(t) \end{pmatrix} = \begin{pmatrix} 0.2 \sin(\frac{\pi}{2}t) \\ -0.7 - 0.01t \\ 1 + 0.2 \cos(\frac{\pi}{2}t) \\ -\frac{\pi}{2} + 0.3 \sin(\frac{\pi}{2}t) \\ \pi \\ \pi \end{pmatrix}, \quad (3.102)$$

dla przedziału czasu  $t \in \mathcal{I} = [0, 10][s]$ . Analiza kinematyki manipulatora, określonej w aspekcie tych współrzędnych zależnością (2.76), prowadzi do

<sup>¶</sup> $\mathbf{V}_{4i} \in \mathbb{R}^i$  — zbiory otwarte.

wniosku, że realizacja trajektorii zadanej wymaga wprowadzenia manipulatora w konfigurację osobliwą wtedy i tylko wtedy, gdy

$$y_{d_1}(t_s) = 0,$$

w chwili  $t_s$ , takiej że  $y_{d_4}(t_s) = \pm\pi/2$ <sup>||</sup>. Dla trajektorii (3.102) warunek ten jest spełniony dla  $t_s \in T = \{0, 2, 4, 6, 8, 10\}[s]$  i powoduje, że zadanie nie może zostać rozwiązane ani korzystając z regularnych metod jacobianowych, ani przy użyciu podejścia algebraicznego<sup>\*\*</sup>.

Widać, że zgodnie z algorytmem metody postaci normalnych powinniśmy powtórzyć jego kroki 2–6 sześć razy, otrzymując tym samym trajektorie przegubową złożoną z jedenastu fragmentów, począwszy od fragmentu osobliwego. Zgodnie z wcześniejszą zapowiedzią postąpimy jednak inaczej. Jak pokazano w poprzednim przykładzie, w konfiguracjach osobliwych rozpatrywana kinematyka jest równoważna hiperbolicznej postaci normalnej (3.101). Aby przy takiej postaci normalnej uzyskane rozwiązanie było ciągłe, należy do jego określenia skorzystać z zależności (3.91), co oznacza, że w konfiguracjach osobliwych rozwiązanie będzie miało postać

$$\mathbf{x}_d(t_s) = \boldsymbol{\varphi}^{-1}\left(\lim_{\tau \rightarrow t_s} \mathbf{r}_0(\boldsymbol{\psi}(\mathbf{y}_d(\tau)))\right). \quad (3.103)$$

Ze względu na gładkość odwzorowania  $\boldsymbol{\varphi}^{-1}(\boldsymbol{\xi})$ , wyrażenie (3.103) można przekształcić do postaci

$$\mathbf{x}_d(t_s) = \lim_{\tau \rightarrow t_s} \boldsymbol{\varphi}^{-1}(\mathbf{r}_0(\boldsymbol{\psi}(\mathbf{y}_d(\tau)))).$$

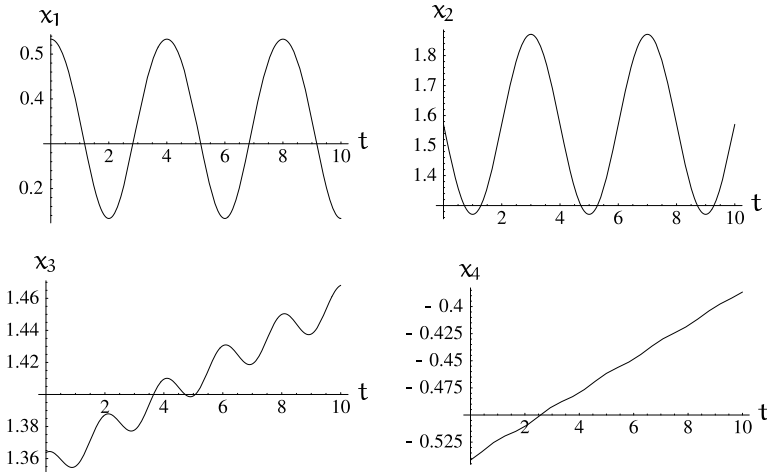
Ponieważ  $\boldsymbol{\varphi}^{-1}(\mathbf{r}_0(\boldsymbol{\psi}(\mathbf{y}_d(\tau))))$  jest rozwiązaniem odwrotnego zadania kinematyki uzyskanym przy użyciu bezpośredniego podejścia algebraicznego, dochodzimy do konkluzji, że w przypadku osobliwym rozwiązanie odwrotnego zadania kinematyki może być określone jako

$$\mathbf{x}_d(t_s) = \lim_{\tau \rightarrow t_s} \mathbf{r}(\mathbf{y}_d(\tau)), \quad (3.104)$$

gdzie  $\mathbf{r}(\mathbf{y})$  jest dane przez (3.15). Trajektorie przegubowe uzyskane przy użyciu formuły (3.104), odpowiadające trajektorii zadaniowej (3.102), przedstawia rysunek 3.30. Nie pokazane na rysunku składowe trajektorie są

<sup>||</sup> Jeśli dla takiego  $t_s$  składowa  $y_{d_1}(t_s)$  trajektorii nie jest równa 0 to znaczy, że trajektoria leży poza przestrzenią roboczą manipulatora.

<sup>\*\*</sup> W tym drugim przypadku, w wyrażeniach opisujących kinematykę odwrotną w osobliwościach pojawia się dzielenie przez zero.



**Rysunek 3.30** Rozwiązanie zadania odwrotnego dla manipulatora IRb-6 zamontowanego na torze jezdnyim uzyskane metodą postaci normalnych.

stałe równe 0. Błędy popełnione podczas obliczeń zawierały się w granicach dokładności prowadzonych obliczeń numerycznych ( $10^{-16}$ ). ■

### 3.2.3 Metoda jacobianu dołączonego

W przypadku punktowego zadania odwrotnego algorytm osobliwej kinematyki odwrotnej powinien być w stanie wyznaczyć konfiguracje osobliwe realizujące określone położenie i orientację w przestrzeni zadaniowej. W tym celu rozważymy najpierw punktowe zadanie odwrotne przy określonym  $\mathbf{y}_d$ , dla kinematyki nieredundantnej  $\mathbf{y} = \mathbf{k}(\mathbf{x})$  o  $n$  stopniach swobody. Przy braku konfiguracji osobliwych do rozwiązania tego zadania stosowaliśmy algorytm Newtona (3.21). Obecnie przedstawimy pewne uogólnienie algorytmu Newtona, zwane *algorytmem Newtona-Smale'a*. Wybierzmy konfigurację początkową  $\mathbf{x}_0$ . Jeżeli  $\mathbf{k}(\mathbf{x}_0) = \mathbf{y}_d$ , to zadanie odwrotne jest rozwiązane. W przeciwnym wypadku założymy, że w każdej konfiguracji jest spełniony *warunek transwersalności*

$$\text{rank} [\mathbf{J}^a(\mathbf{x}) \quad \mathbf{k}(\mathbf{x}_0) - \mathbf{y}_d] = n. \quad (3.105)$$

Warunek ten jest w oczywisty sposób spełniony w konfiguracjach regularnych. Natomiast, jeżeli  $\mathbf{x}$  oznacza konfigurację osobliwą, to warunek transwersalności wymaga, by wektor  $\mathbf{k}(\mathbf{x}_0) - \mathbf{y}_d$  uzupełniał przestrzeń generowaną przez kolumny jacobianu analitycznego  $\mathbf{J}^a(\mathbf{x})$  do przestrzeni zadaniowej.

wej  $\mathbb{R}^n$ , zatem był transwersalny do podprzestrzeni rozpiętej przez kolumny tej macierzy w konfiguracji  $\mathbf{x}$ . W konfiguracji osobliwej spełnienie warunku transwersalności pociąga za sobą wymaganie, aby korząd konfiguracji był równy 1. Dla takich konfiguracji warunek transwersalności jest możliwy do spełnienia dzięki odpowiedniemu doborowi konfiguracji początkowej lub docelowych współrzędnych zadaniowych. Przy spełnionym warunku (3.105) algorytm Newtona-Smale'a jest zdefiniowany za pośrednictwem układu równań różniczkowych

$$\dot{\mathbf{x}} = -\alpha \operatorname{adj} \mathbf{J}^a(\mathbf{x})(\mathbf{k}(\mathbf{x}) - \mathbf{y}_d), \quad (3.106)$$

z warunkiem początkowym  $\mathbf{x}_0$ , w którym  $\operatorname{adj} \mathbf{J}^a(\mathbf{x})$  oznacza macierz dołączoną jakobianu analitycznego  $\mathbf{J}^a(\mathbf{x})^*$ . Przypominamy, że macierz dołączona posiada następującą własność

$$(\operatorname{adj} \mathbf{J}^a) \mathbf{J}^a = \mathbf{J}^a \operatorname{adj} \mathbf{J}^a = \det \mathbf{J}^a \mathbb{I}_n.$$

Znak współczynnika  $\alpha$  w algorytmie (3.106) jest taki jak znak wyznacznika jakobianu w punkcie startowym algorytmu,  $\operatorname{sgn} \alpha = \operatorname{sgn} \det \mathbf{J}^a(\mathbf{x}_0)$ . Algorytm Newtona-Smale'a zapewnia asymptotyczne rozwiązanie punktowego zadania odwrotnego w tym sensie, że  $\lim_{t \rightarrow +\infty} \mathbf{x}(t) = \mathbf{x}_d$  oraz  $\mathbf{y}_d = \mathbf{k}(\mathbf{x}_d)$ . Algorytm ten można rozszerzyć na punktowe zadanie odwrotne dla kinematyki redundantnej  $\mathbf{y} = \mathbf{k}(\mathbf{x})$ ,  $m > n$ . Wybierzmy konfigurację początkową  $\mathbf{x}_0$  i niech  $\mathbf{y}_d$  oznacza docelowe współrzędne zadaniowe. Przyjmijmy warunek transwersalności postaci

$$\operatorname{rank} [\mathbf{J}^a(\mathbf{x}) \mathbf{J}^{aT}(\mathbf{x}) \quad \mathbf{k}(\mathbf{x}_0) - \mathbf{y}_d] = m. \quad (3.107)$$

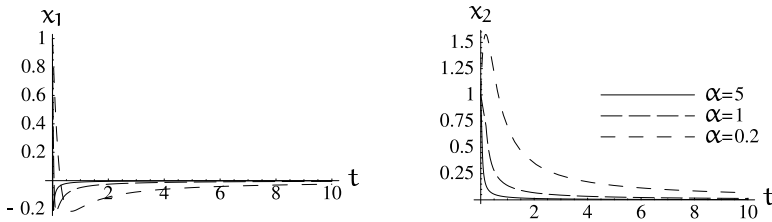
Biorąc to pod uwagę, rozwiązanie zadania odwrotnego otrzymuje się jako granicę przy  $t \rightarrow +\infty$  trajektorii  $\mathbf{x}(t)$  układu

$$\dot{\mathbf{x}} = -\alpha \mathbf{J}^{aT}(\mathbf{x}) \operatorname{adj} (\mathbf{J}^a(\mathbf{x}) \mathbf{J}^{aT}(\mathbf{x})) (\mathbf{k}(\mathbf{x}) - \mathbf{y}_d), \quad (3.108)$$

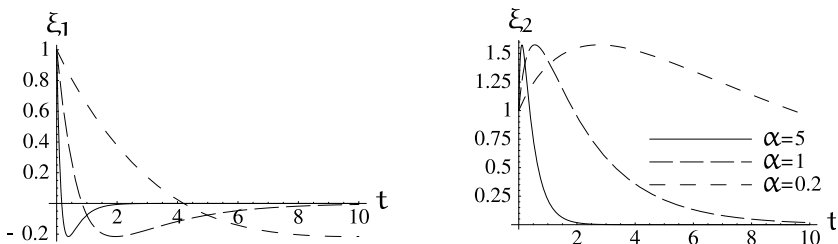
zainicjowanego w  $\mathbf{x}_0$ . We wzorze (3.108)  $\operatorname{adj}$  ponownie oznacza macierz dołączoną, natomiast znak współczynnika  $\alpha$  jest taki jak  $\operatorname{sgn} \det (\mathbf{J}^a(\mathbf{x}_0) \mathbf{J}^{aT}(\mathbf{x}_0))$ . Analiza działania algorytmu Newtona-Smale'a pokazuje, że w wyniku realizacji trajektorii będącej rozwiązaniem układu (3.106) lub (3.108), ruch w przestrzeni zadaniowej odbywa się wzdłuż linii prostej  $\mathbf{y} = \mathbf{y}_0 + s(\mathbf{y}_d - \mathbf{y}_0)$ , przy czym rozwiązanie zadania odwrotnego odpowiada wartości parametru  $s = 1$ . W efekcie, w przestrzeni zadaniowej zostaje zdefiniowany pewien

\*Zobacz dodatek A.1.





Rysunek 3.31 Trajektorie przegubowe uzyskane przy zastosowaniu algorytmu Newtona-Smale'a.



Rysunek 3.32 Trajektorie przegubowe uzyskane przy zastosowaniu algorytmu Newtona.

prostoliniowy tor (ścieżka) ruchu, a celem algorytmu kinematyki odwrotnej jest zapewnienie ruchu wzdłuż tego toru. Dynamika ruchu wzdłuż toru ma znaczenie drugorzędne.

### Przykład 3.2.7 (Manipulator typu podwójne wahadło)

Rozpatrzmy manipulator typu podwójne wahadło, którego ramiona mają długość  $l_1 = 5$ ,  $l_2 = 3$ , opisany kinematyką (2.68). Niech naszym zadaniem będzie określenie konfiguracji manipulatora, przy której jego efektor osiąga punkt  $\mathbf{y}_d = (8, 0)^T$  trajektorii (3.94) dla  $t = 2.5$ . Jak wiemy z poprzednich przykładów, w tym celu manipulator musi przyjąć konfigurację osobliwą. Efekt zastosowania algorytmu Newtona-Smale'a (3.106) zainicjowanego w punkcie  $\mathbf{x}_0 = (1, 1)^T$ , przy różnych wartościach parametru  $\alpha$ , przedstawia rysunek 3.31.

Nieco zaskakującym wydaje się spostrzeżenie, że postawione wyżej zadanie można także rozwiązać stosując algorytm Newtona (3.21), pomimo że uwarunkowanego układu równań opisujących ten algorytm (osobliwa macierz  $\mathbf{J}^a(\boldsymbol{\xi})$ ). Komplet wyników analogicznych do przedstawionych na rysunku 3.31, uzyskany przy pomocy algorytmu Newtona, pokazuje rysunek 3.32. ■

### 3.2.4 Metoda przestrzeni zerowej

Gdyby pójść dalej za tokiem rozumowania przedstawionym w poprzednim podrozdziale i dopuścić jako rozwiązanie odwrotnego zadania kinematyki każdy ruch w przestrzeni zadaniowej wzdłuż zadanego toru  $\mathbf{y}_d(s)$ , bez określania dynamiki tego ruchu, to możliwości rozwiązania osobliwego zadania odwrotnego wzrosną. Niech zatem będzie dany manipulator nieredundantny z kinematyką  $\mathbf{y} = \mathbf{k}(\mathbf{x})$ . Sformułujmy odwrotne zadanie kinematyki w następujący sposób:

Dana jest krzywa  $\mathbf{y}_d(s)$  w przestrzeni zadaniowej, znaleźć trajektorię ruchu przegubów  $\mathbf{x}_d(t)$  i dobrać dynamikę ruchu wzdłuż krzywej  $\mathbf{y}_d(s)$  (tzn. zależność parametru od czasu), tak aby

$$\mathbf{y}_d(s(t)) = \mathbf{k}(\mathbf{x}_d(t)). \quad (3.109)$$

Aby rozwiązać tak rozszerzone zadanie, potraktujemy  $s$  jako dodatkową współrzędną przegubową i zdefiniujemy odwzorowanie

$$\mathbf{F}(\mathbf{x}, s) = \mathbf{k}(\mathbf{x}) - \mathbf{y}_d(s). \quad (3.110)$$

Zakładając, że istnieje rozwiązanie  $(\mathbf{x}_d(t), s(t))$  zadania odwrotnego, obliczamy pochodną lewej strony wyrażenia (3.110) względem czasu. Na mocy zależności (3.109)

$$\left[ \frac{\partial \mathbf{F}(\mathbf{x}, s)}{\partial \mathbf{x}} \quad \frac{\partial \mathbf{F}(\mathbf{x}, s)}{\partial s} \right] \begin{pmatrix} \dot{\mathbf{x}} \\ \dot{s} \end{pmatrix} = \mathbf{0},$$

skąd wynika, że trajektoria  $(\mathbf{x}_d(t), s(t))$  stanowi krzywą całkową układu dynamicznego ruchu własnego dla kinematyki (3.110)

$$\begin{pmatrix} \dot{\mathbf{x}}_d \\ \dot{s} \end{pmatrix} = \mathbf{X}(\mathbf{x}_d, s). \quad (3.111)$$

Pole wektorowe ruchu własnego  $\mathbf{X}(\mathbf{x}, s) \in \text{Ker} \begin{bmatrix} \frac{\partial \mathbf{F}}{\partial \mathbf{x}} & \frac{\partial \mathbf{F}}{\partial s} \end{bmatrix}$  definiuje się w sposób kanoniczny formułą (3.59), a zatem

$$\mathbf{X}_i(\mathbf{x}, s) = (-1)^{i+1} \det \left[ \frac{\partial \mathbf{F}(\mathbf{x}, s)}{\partial \mathbf{x}} \quad \frac{\partial \mathbf{F}(\mathbf{x}, s)}{\partial s} \right]^i, \quad (3.112)$$

gdzie wyznacznik jest obliczany po usunięciu  $i$ -tej kolumny z macierzy Jacobiego odwzorowania  $F(\mathbf{x}, s)$ . Zauważmy, że ostatnia składowa pola wektorowego ruchu własnego,  $\mathbf{X}_{n+1}(\mathbf{x}, s) = \det \mathbf{J}^a(\mathbf{x})$ , co oznacza, że dynamika ruchu wzdłuż krzywej  $\mathbf{y}_d(s)$  jest zdefiniowana przez układ

$$\dot{s} = \det \mathbf{J}^a(\mathbf{x}_d). \quad (3.113)$$

Widzimy, że efektor zatrzymuje się w obrazach konfiguracji osobliwych. Jeżeli dodatkowo przyjąć warunek transwersalności

$$\text{rank} \left[ \mathbf{J}^a(\mathbf{x}) \quad \frac{d\mathbf{y}_d(s)}{ds} \right] = n, \quad (3.114)$$

to w każdym punkcie  $\mathbf{X}(\mathbf{x}, s) \neq 0$ , a zatem trajektoria w przestrzeni przegubowej przechodzi przez konfiguracje osobliwe bez zatrzymania.

### Przykład 3.2.8 (Manipulator typu podwójne wahadło)

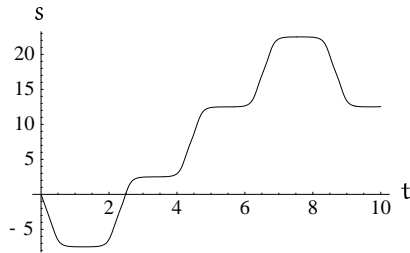
Jak wiemy, metoda przestrzeni zerowej pozwala na rozwiązanie osobliwego odwrotnego zadania kinematyki przy danej ścieżce w przestrzeni zadaniowej. Przyjmijmy więc, że zadanie manipulatora polega na wykonaniu ruchu efektora wzdłuż ścieżki w kształcie okręgu\*

$$\mathbf{y}_d(s) = \begin{pmatrix} y_{d1}(s) \\ y_{d2}(s) \end{pmatrix} = \begin{pmatrix} 6 + 2 \sin\left(\frac{\pi}{5}s\right) \\ 2 \cos\left(\frac{\pi}{5}s\right) \end{pmatrix}. \quad (3.115)$$

Ponownie, ścieżkę dobrano tak, aby zawierała konfiguracje osobliwe (tutaj dla  $s \bmod 10 = 2.5$ ). W rezultacie zastosowania metody przestrzeni zerowej otrzymujemy parametryzację czasową  $s(t)$  zadanej ścieżki oraz trajektorię przegubową  $\mathbf{x}_d(t)$  odpowiadającą trajektorii zadaniowej  $\mathbf{y}_d(s(t))$ . Przebiegi te, wynikające z rozwiązania układu równań (3.111) zainicjowanych w punkcie  $\mathbf{x}_0 = (0.805, -1.37)^\top$ ,  $s_0 = 0$ , przedstawiają rysunki 3.33 i 3.34.

Jak widać z rysunków, w przedziale czasu  $\mathcal{I} = [0, 10]$  efektor manipulatora wykonał niemal pięć obrotów. Każdorazowo, gdy manipulator zbliżał się do konfiguracji osobliwej, zatrzymywał się. Symulacje pokazują, że w metodzie nie mamy wpływu ani na to, czy manipulator przejdzie przez konfigurację osobliwą (czemu na skutek zmiany wyznacznika jacobianu manipulatora zawsze towarzyszy zmiana kierunku realizacji ścieżki — zobacz równanie (3.113)), czy tylko ulegnie odbiciu od niej, ani na to, w którą stronę zachodzi ruch (funkcja  $s(t)$  nie musi być monotoniczna). W przedstawionym

\*Której odpowiada między innymi trajektoria (3.94).



**Rysunek 3.33** Parametryzacja czasowa ścieżki uzyskana metodą przestrzeni zerowej.

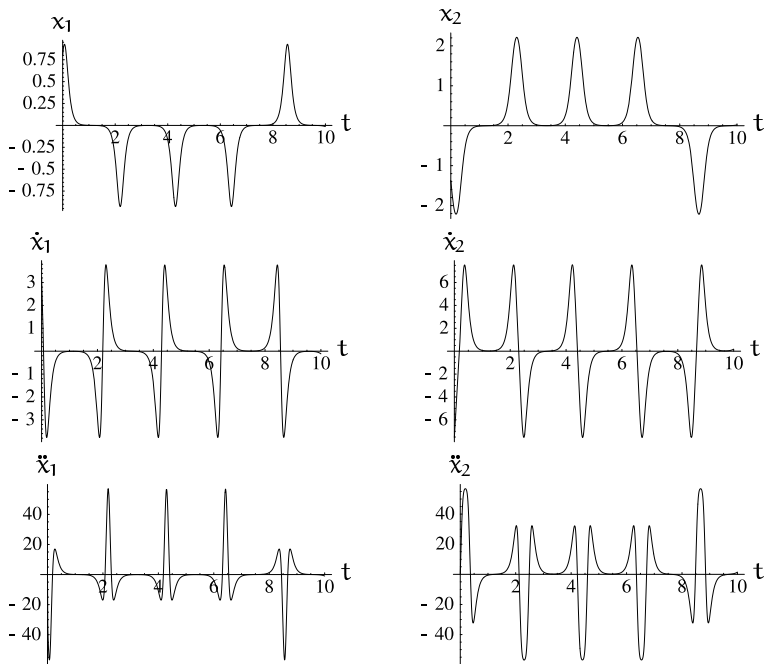
przypadku trajektoria przegubowa najpierw przechodzi przez osobliwość w chwili  $t = 1.24$ , następnie doznaje odbicia (w dwóch chwilach  $t = 3.35$ ,  $t = 5.47$ ) i przechodzi ponownie (przy  $t = 7.60$ ). Zauważmy, że z dala od konfiguracji osobliwych uzyskane rozwiązania charakteryzują się dużymi prędkościami i bardzo dużymi przyspieszeniami (porównaj rozwiązania metodą postaci normalnych — przykłady 3.2.3 i 3.2.4, oraz metodą jakobianu odpornego — przykład 3.2.9). Błędy towarzyszące rozwiązaniu metodą przestrzeni zerowej pokazuje rysunek 3.35. ■

### 3.2.5 Metoda jakobianu odpornego

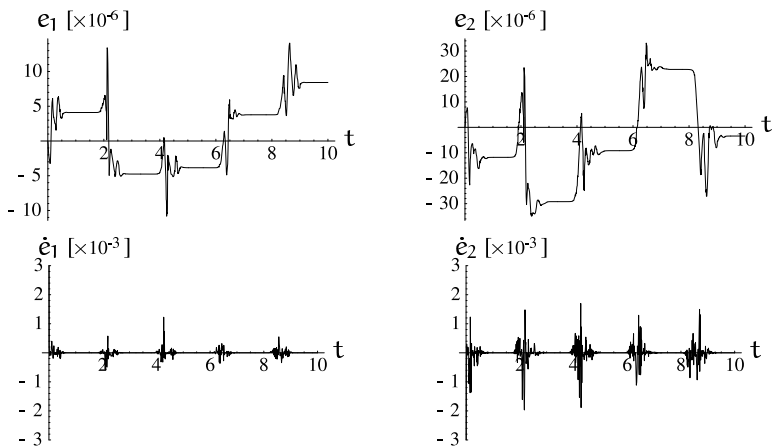
W praktyce często wystarczające jest uzyskanie przybliżonego rozwiązania zadania odwrotnego. Dominującą metodą przybliżoną jest metoda najmniejszych kwadratów pozwalająca na otrzymanie algorytmów analogicznych do (3.20), (3.28), (3.33), dobrze określonych także w pobliżu konfiguracji osobliwych. Istotą metody najmniejszych kwadratów jest zastąpienie odwrotności lub pseudoodwrotności jakobianu analitycznego, źle uwarunkowanego w konfiguracjach osobliwych, odwrotnością przybliżoną, lecz uodpornioną na obecność osobliwości. *Odwrotność przybliżoną* wyznacza się przez minimalizację kwadratowej funkcji jakości

$$V(\mathbf{v}) = \frac{1}{2} \lambda \mathbf{v}^T \mathbf{v} + \frac{1}{2} (\mathbf{w} - \mathbf{J}^a(\mathbf{x})\mathbf{v})^T (\mathbf{w} - \mathbf{J}^a(\mathbf{x})\mathbf{v}),$$

będącej sumą ważoną energii ruchu w przegubach oraz kwadratu błędu prędkości, w której  $\lambda > 0$  jest wagą składnika energii. Po wykonaniu odpo-



**Rysunek 3.34** Trajektorie położenia, prędkości i przyspieszenia przegubów używane przy zastosowaniu metody przestrzeni zerowej.



**Rysunek 3.35** Błędy położenia i prędkości efektora przy metodzie przestrzeni zerowej.

wiednich przekształceń wyliczamy

$$\begin{aligned} \mathbf{J}^{a\lambda}(\mathbf{x}) &= (\lambda \mathbb{I}_n + \mathbf{J}^{aT}(\mathbf{x})\mathbf{J}^a(\mathbf{x}))^{-1} \mathbf{J}^{aT}(\mathbf{x}) = \\ &= \mathbf{J}^{aT}(\mathbf{x}) (\lambda \mathbb{I}_m + \mathbf{J}^a(\mathbf{x})\mathbf{J}^{aT}(\mathbf{x}))^{-1}. \end{aligned} \quad (3.116)$$

Zauważmy, że druga z formuł w wyrażeniu (3.116) przy  $\lambda = 0$  staje się prawostronną pseudoodwrotnością  $\mathbf{J}^{a\#}(\mathbf{x})$  jakobianu analitycznego (zdefiniowanego dla macierzy  $\mathbf{W} = \mathbb{I}_n$ ). W ogólności, rola parametru  $\lambda$  polega na zapewnieniu istnienia macierzy odwrotnych definiujących  $\mathbf{J}^{a\lambda}(\mathbf{x})$  w konfiguracjach osobliwych manipulatora. Ponieważ przy  $\lambda > 0$

$$\mathbf{J}^a(\mathbf{x})\mathbf{J}^{a\lambda}(\mathbf{x}) \neq \mathbb{I}_m \quad \text{oraz} \quad \mathbf{J}^{a\lambda}(\mathbf{x})\mathbf{J}^a(\mathbf{x}) \neq \mathbb{I}_n,$$

rozwiązanie zadania odwrotnego przy zastosowaniu odwrotności  $\mathbf{J}^{a\lambda}(\mathbf{x})$  jest przybliżone. Zastępując w formułach (3.20), (3.28)  $(\mathbf{J}^a)^{-1}(\mathbf{x})$  lub  $\mathbf{J}^{a\#}(\mathbf{x})$  przez  $\mathbf{J}^{a\lambda}(\mathbf{x})$  otrzymujemy następujący *algorytm jakobianu odpornego*

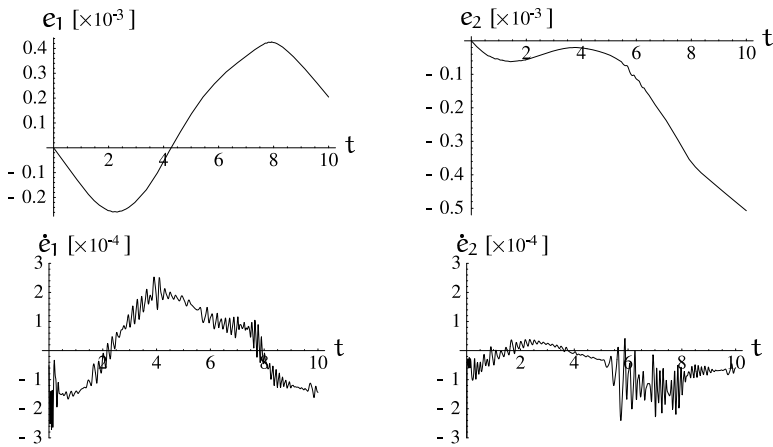
$$\dot{\mathbf{x}}_d = \mathbf{J}^{a\lambda}(\mathbf{x}_d)\dot{\mathbf{y}}_d. \quad (3.117)$$

Przy założeniu, że konfiguracja początkowa  $\mathbf{x}_0$  tego algorytmu jest nieosobliwa, możemy do jej wyznaczenia posłużyć się algorytmem (3.21) lub (3.29). Uodporniona na osobliwości wersja algorytmu (3.33) ma postać

$$\dot{\mathbf{x}}_d = \mathbf{J}^{a\lambda}(\mathbf{x}_d)\dot{\mathbf{y}}_d + (\mathbb{I}_n - \mathbf{J}^{a\lambda}(\mathbf{x}_d)\mathbf{J}^a(\mathbf{x}_d))\mathbf{v}. \quad (3.118)$$

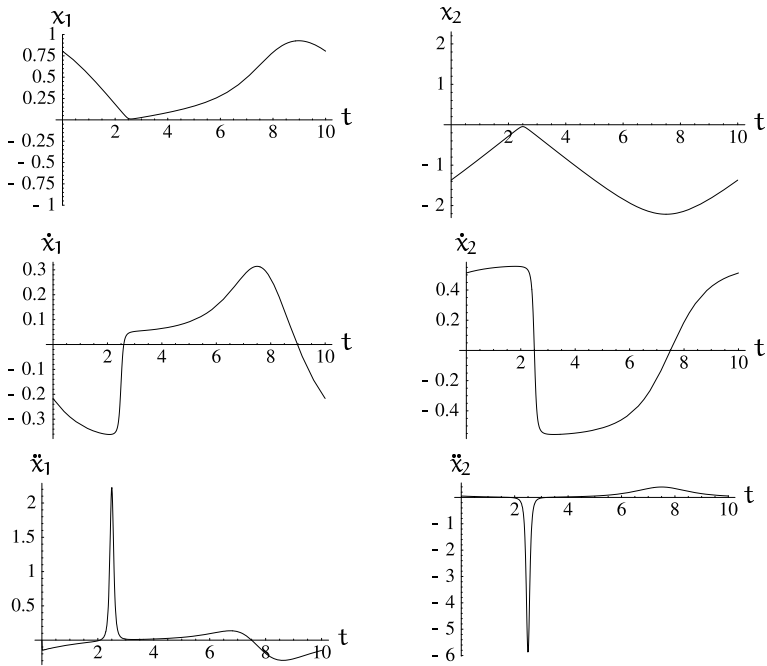
### Przykład 3.2.9 (Manipulator typu podwójne wahadło)

Aby przekonać się, jakie są skutki uodpornienia na osobliwości metody jakobianu odwrotnego, rozwiążemy najpierw metodą jakobianu odpornego zadanie postawione w przykładzie 3.1.5. Niech efektor manipulatora ma podążać wzdłuż trajektorii zadaniowej (3.23). Rozwiązanie tak postawionego zadania uzyskane po scałkowaniu równań (3.117) z parametrem  $\lambda = 0.001$ , zainicjowanych w punkcie  $\mathbf{x}_0 = (-0.201, 1.74)^T$ , wygląda podobnie do rozwiązania przedstawionego na rysunku 3.4. Różnica pojawia się natomiast w przebiegach błędów położenia i prędkości efektora. W tym przypadku, oprócz błędów numerycznych, występują błędy metody, co wyraźnie widać na rysunku 3.36 (porównaj z rysunkiem 3.5). Prowadzi to do zwiększenia sumarycznych błędów o ponad jeden rząd wielkości. Dodajmy, że czas obliczeń wzrósł w podobnym stosunku.

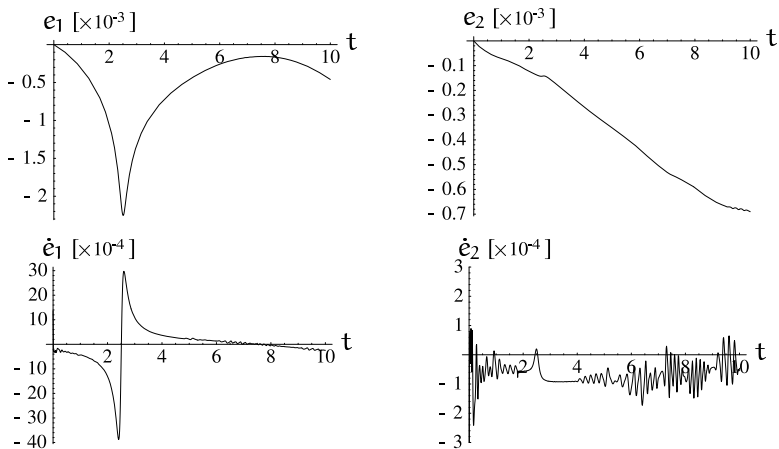


**Rysunek 3.36** Błędy położenia i prędkości efektora towarzyszące trajektorii regularnej przy algorytmie jacobianu odpornego.

Przejdźmy teraz do rozwiązania odwrotnego zadania kinematyki metodą jacobianu odpornego w sytuacji, gdy trajektoria zadana wymaga osiągnięcia konfiguracji osobliwej. W tym celu zażądajmy, aby efektor manipulatora podążał wzdłuż trajektorii zadaniowej (3.94). Rozwiązując to zadanie z wykorzystaniem równań (3.117) z parametrem  $\lambda = 0.001$ , zainicjowanych w punkcie  $\mathbf{x}_0 = (0.805, -1.37)^T$ , otrzymujemy trajektorie przedstawione na rysunku 3.37. Porównanie tych trajektorii z rozwiązaniami zadania uzyskanymi dotychczas (zobacz rysunki 3.28 i 3.29) prowadzi do wniosku, że są one podobne do rozwiązania otrzymanego bezpośrednio metodą algebraiczną. Różnice między tymi rozwiązaniami dotyczą wartości przyspieszeń w przegubach manipulatora i wielkości błędów. Dzięki zastosowaniu metody jacobianu odpornego, przyspieszenia zostały zredukowane z wartości teoretycznie nieskończonej do poziomu pojedynczych  $\text{rad/s}^2$  (jest to nadal znaczna wartość w porównaniu z przyspieszeniami wzdłuż analogicznych trajektorii wynikającymi z metody postaci normalnych). Redukcja ta została okupiona wielkością powstałych błędów, przedstawionych na rysunku 3.38, które w tym przypadku przyjmują znaczące wartości (o dwa rzędy większe od błędów podstawowej metody jacobianowej). Cechą charakterystyczną uzyskanego rozwiązania jest to, że maksymalne wartości błędów pojawiają się w chwilach, gdy trajektoria zadana osiąga konfiguracje osobliwe. Oczywiście, błędy te można zmniejszyć przez zmniejszenie wartości parametru  $\lambda$ , ale towarzyszyć temu będzie wzrost przyspieszeń w przegubach manipulatora w pobliżu konfiguracji osobliwych. ■

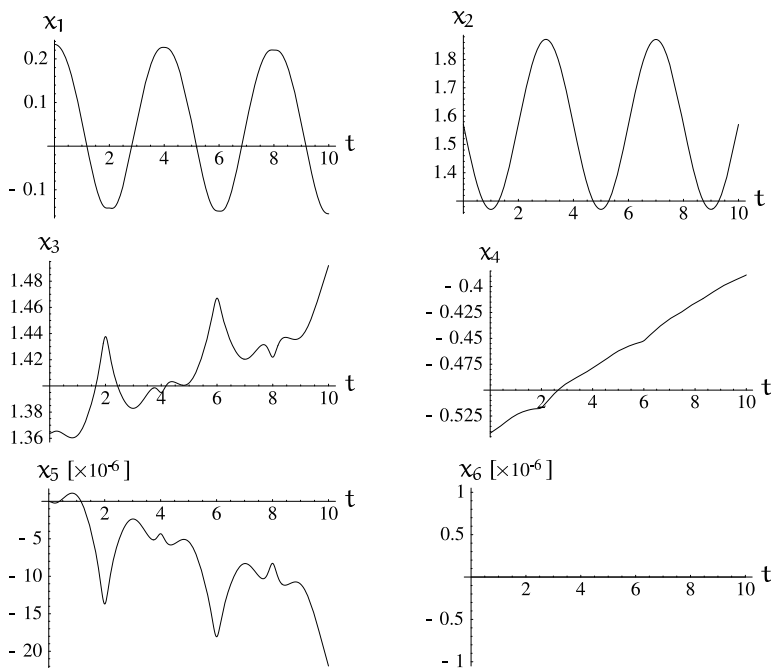


**Rysunek 3.37** Rozwiązanie osobliwego zadania odwrotnego dla podwójnego wahadła uzyskane przy pomocy algorytmu jacobianu odpornego.



**Rysunek 3.38** Błędy położenia i prędkości efektora towarzyszące trajektorii osobliwej przy algorytmie jacobianu odpornego.





**Rysunek 3.39** Rozwiązanie zadania odwrotnego dla manipulatora IRb-6 zamontowanego na torze jezdnyim uzyskane przy pomocy algorytmu jakobianu odpornego.

### Przykład 3.2.10 (Manipulator IRb-6 na torze jezdnyim)

W tym przykładzie ponownie przyjmujemy, że zadaniem manipulatora IRb-6 zamontowanego na torze jezdnyim jest przemieszczenie efektora wzdłuż trajektorii (3.102). Rozwiązanie tego zadania metodą jakobianu odpornego przedstawia rysunek 3.39. Uzyskane rozwiązanie jest bardzo podobne do rozwiązania możliwego do osiągnięcia metodą postaci normalnych. Różnice dotyczą dwóch szczegółów. Po pierwsze, błędy metody jakobianu odpornego kształtują się na poziomie  $10^{-3}$ , podczas gdy metodzie postaci normalnych towarzyszą jedynie błędy numeryczne rzędu  $10^{-16}$ . Po drugie, znalezienie rozwiązania metodą jakobianu odpornego było ponad 1000-krotnie bardziej czasochłonne niż rozwiązanie zadania metodą postaci normalnych. Na domiar złego, przy wielokrotnym powtarzaniu trajektorii zadanej błędy w metodzie jakobianu odpornego kumulują się. ■

### 3.3 Komentarze i uwagi bibliograficzne

Klasyczną definicję przestrzeni roboczej manipulatora podają na przykład prace [Lat93, SS96]. Odwrotne zadanie kinematyki manipulatora należy do podstawowych zadań robotyki [MLS94, SV97, SS96, Ang97, CD98]. Z regularnym zadaniem odwrotnym mamy do czynienia wtedy, gdy w celu jego rozwiązania nie ma potrzeby wprowadzania manipulatora w konfiguracje osobliwe. Jak zaznaczyliśmy, występowanie konfiguracji osobliwych jest konsekwencją odmiennej geometrii rozmaitości przegubowej i rozmaitości zadaniowej [Got86, BW88]. Systematyczną technikę rozwiązania odwrotnego zadania kinematyki w postaci jawnej, bazującą na wykładniczej reprezentacji kinematyki, stanowi dekompozycja zadania odwrotnego na tzw. podproblemy Padena-Kahana, [MLS94]. Przy omawianiu algorytmów kinematyki odwrotnej kładziemy szczególny nacisk na metody jakobiano-  
we [MLS94, SV97, SS96]. Wykorzystanie pseudoodwrotności jakobianu analitycznego do rozwiązania zadania odwrotnego w przypadku kinematyki redundantnej omówiono szczegółowo w monografii [Nak91]. Koncepcję wykorzystania formuły (3.33) do kształtowania przebiegu trajektorii przegubów stosownie do założonych kryteriów sformułował Liégeois [Lié77]. Funkcja odległości przedstawiona wzorem (3.36) została zaczerpnięta z książki [SS96]. Algorytm kinematyki odwrotnej wykorzystujący pojęcie jakobianu rozszerzonego podał Bailleul [Bai85]. Algorytm (3.43) pochodzi od Siciliano [SS96]. Wadą algorytmów redundantnej kinematyki odwrotnej wykorzystujących pseudoodwrotność jakobianu jest brak *powtarzalności* [KH83], która polega na tym, że zamkniętym trajektoriom (torom) w przestrzeni zadaniowej odpowiadają zamknięte trajektorie (tory) w przestrzeni przegubowej. Kryteria powtarzalności zostały sformułowane przez Bocketta [Bro84], a następnie rozwinięte przez Shamira i Yomdina [SY88]. Powtarzalnymi algorytmami kinematyki odwrotnej zajmował się Maciejewski [RM94]. Od wady braku powtarzalności jest wolny algorytm kinematyki odwrotnej oparty na zastosowaniu mnożników Lagrange'a, zaproponowany przez Changa [Cha87]. Elipsoida manipulowalności i miara manipulowalności (3.53) są autorstwa Yoshikawy [Yos84, Yos91]. Manipulowalność należy do podstawowych miar zręczności manipulatora. Zagadnieniu miar zręczności i fundamentom projektowania zręcznych manipulatorów poświęcono prace [KB87, PB94, SB96]. Na możliwość unikania konfiguracji osobliwych przy sterowaniu ruchem manipulatorów redundantnych zwrócili uwagę Bailleul, Hollerbach i Bockett [BHB84]. Zagadnienia unikania osobliwości dotyczą prace [Bed91,

BF91] i [Sha90, Bei97, SOC97]. Własności „konfiguracja lokalnie możliwa do uniknięcia” i „konfiguracja lokalnie niemożliwa do uniknięcia” wraz z kryteriami tych własności zakorzenionymi w teorii hamiltonowskich układów dynamicznych i w postaciach normalnych kinematyki zostały wprowadzone w pracach [TM95, Tch95b, Tch97, Tch98]. Twierdzenie 3.2.1 pochodzi z pracy [TM95]. Pojęcie rozmaitości ruchu własnego zostało wprowadzone przez Burdicka [Bur89, Bur91]. Hamiltonowskie pola wektorowe zdefiniowano w książce [AGLV93]. Twierdzenie 3.2.2 zostało wykazane w artykule [Tch95b]. Do automatycznej analizy osobliwości kinematyki według kryteriów przedstawionych w podrozdziale 3.2.1 służy program KOSMA, opisany w pracy [Piw98]. Idea klasyfikacji kinematyki manipulatorów przy pomocy prostych modeli matematycznych zwanych postaciami normalnymi należy do standardowych narzędzi teorii osobliwości odwzorowań, [GG73, AVZ85, Arn93]. Pierwsze próby takiej klasyfikacji można znaleźć w pracach [Tch90, Tch91]. Uzyskane później wyniki dotyczą głównie warunków transformacji kinematyki nieredundantnej i redundantnej do kwadratowej (parabolicznej) i hiperbolicznej postaci normalnej [Tch95a, MT96, Tch98]. Twierdzenie 3.2.3 pochodzi z pracy [Tch98]. Twierdzenie 3.2.4 prezentujemy po raz pierwszy w niniejszej książce. Jest ono naturalną konsekwencją wcześniejszych wyników cząstkowych [Mus96, TMZ97, Tch99]. Podstawy metody postaci normalnych, polegającej na rozwiązaniu osobliwego odwrotnego zadania kinematyki przez jego redukcję do równoważnego zadania dla postaci normalnej kinematyki przedstawiono w pracach [Mus96, TM97, TM98]. Metoda postaci normalnych została poddana weryfikacji eksperymentalnej na manipulatorze *AdeptOne\**, [TM98], oraz na manipulatorze IRb-6 zamontowanym na torze jezdny [MT98]. Wykorzystanie idei geometrii i topologii różniczkowej do badania kinematyki manipulatorów, bliskie podejściu zastosowanemu w metodzie postaci normalnych, zawierają m. in. prace Kieffera [Kie92, Kie94], Pai i Leu [PL92], Shankara i Sarafa [SS95] oraz Chena, O’Neila i Senga [CSO97, OCS97, SOC97]. Podstawy metody rozwiązania zadania kinematyki odwrotnej, znanej obecnie jako metoda jakobianu dołączonego, zostały podane w pracy [TD93], w której zaproponowano zastosowanie do osobliwego odwrotnego zadania kinematyki manipulatora tzw. uogólnionego algorytmu Newtona, pochodzącego od Smale’a, [Sma76] i nazwanego w podrozdziale 3.2.3 algorytmem Newtona-

---

\*Za umożliwienie przeprowadzenia tych badań jesteśmy zobowiązani Prof. W. Jacakowi, Uniwersytet Johannesesa Keplera, Linz, Austria.

Smale'a. Metoda została rozwinięta przez Nencheva i współpracowników w pracach [NU97, T<sup>+</sup>97, NTU98]. Metoda przestrzeni zerowej pochodzi od Nencheva [Nen95]; metoda nawiązuje do pewnych idei sformułowanych przez Sampei i Furutę [SF88] oraz Kieffera [Kie92, Kie94]. Zastosowanie metody najmniejszych kwadratów do uzyskania przybliżonego algorytmu kinematyki odwrotnej zaproponowali Nakamura i Hanafusa [NH86, Nak91] oraz Wampler II [Wam86]. Szczegółowa analiza tego sposobu podejścia została przeprowadzona w pracy [CSE94]. Problematyka osobliwości kinematyki manipulatorów i algorytmów osobliwej kinematyki odwrotnej stanowi obszar znacznej aktywności badawczej. Jej celem jest opracowanie algorytmów zapewniających dużą dokładność przejścia przez konfiguracje osobliwe i zarazem nieskomplikowanych pojęciowo i obliczeniowo. W tym kontekście niedostatkami metody postaci normalnych jest konieczność rozpoznania typu osobliwości przed zastosowaniem algorytmu.

## Literatura

- [AGLV93] V. I. Arnold, V. V. Goryunov, O. V. Lyashko i V. A. Vasilev, Singularity theory II. classification and applications. W: V. I. Arnold, (red.), *Dynamical Systems VIII*. Springer-Verlag, Berlin, 1993.
- [Ang97] J. Angeles, *Fundamentals of Robotic Mechanical Systems. Theory, Methods, and Algorithms*. Springer-Verlag, New York, 1997.
- [Arn93] V. I. Arnold, (red.), *Dynamical Systems VI. Singularity Theory I*. Springer-Verlag, Berlin, 1993.
- [AVGZ85] V. I. Arnold, A. N. Varchenko i S. M. Gusein-Zade, *Singularities of Differentiable Maps*. Birkhäuser, Boston, 1985.
- [Bai85] J. Bailleul, Kinematic programming alternatives for redundant manipulators. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 722–728, St. Louis, 1985.
- [Bed91] N. S. Bedrossian, Classification of singular configurations for redundant manipulators. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 818–823, Sacramento, 1991.
- [Bei97] L. Beiner, Singularity avoidance for articulated robots. *Robotics Autonomous Systems*, 20:39–47, 1997.
- [BF91] N. S. Bedrossian i K. Flueckiger, Characterizing spatial redundant manipulator singularities. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 714–719, Sacramento, 1991.

- [BHB84] J. Bailleul, J. Hollerbach i R. W. Brockett, Programming and control of kinematically redundant manipulators. W: *Proc. IEEE Conf. Decision Control*, vol. 1, strony 768–774, 1984.
- [Bro84] R. W. Brockett, Robotic manipulators and the product of exponentials formula. W: *Mathematical Theory of Networks and Systems*, strony 120–129. Springer-Verlag, Berlin, 1984.
- [Bur89] J. W. Burdick, On the inverse kinematics of redundant manipulators: Characterization of the self-motion manifolds. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 264–270, Scottsdale, 1989.
- [Bur91] J. W. Burdick, A classification of 3R regional manipulator singularities and geometries. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 3, strony 2670–2675, Sacramento, 1991.
- [BW88] D. R. Baker i C. W. Wampler II, On the inverse kinematics of redundant manipulators. *Int. J. Robotics Research*, 7(2):3–21, 1988.
- [CD98] C. D. Crane III i J. Duffy, *Kinematic Analysis of Robot Manipulators*. Cambridge University Press, Cambridge, 1998.
- [Cha87] P. H. Chang, Closed-form solution for inverse kinematics of robot manipulators with redundancy. *J. Robotic Automat.*, 3(5):383–403, 1987.
- [CSE94] S. Chiaverini, B. Siciliano i O. Egeland, Review of the damped least-squares inverse kinematics with experiments on an industrial robot manipulator. *IEEE Trans. Control Syst. Technology*, 2(2):123–134, 1994.
- [CSO97] Y. C. Chen, J. Seng i K. A. O’Neil, Lowest-order rate control of mechanisms near singularities. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 3, strony 3597–3601, Albuquerque, 1997.
- [GG73] M. Golubitsky i V. Guillemin, *Stable Mappings and Their Singularities*. Springer-Verlag, Berlin, 1973.
- [Got86] D. H. Gottlieb, Robots and fibre bundles. *Bull. Soc. Math. Belgique*, 37:219–223, 1986.
- [KB87] C. A. Klein i B. E. Blaho, Dexterity measures for the design and control of kinematically redundant manipulators. *Int. J. Robotics Research*, 6(2):72–83, 1987.
- [KH83] C. Klein i C. Huang, Review of pseudoinverse control for use with kinematically redundant manipulators. *IEEE Trans. Systems, Man, Cybernet.*, 13(2):245–250, 1983.
- [Kie92] J. Kieffer, Manipulator inverse kinematics for untimed end-effector trajectories with ordinary singularities. *Int. J. Robotics Research*, 11(3):225–237, 1992.

- [Kie94] J. Kieffer, Differential analysis of bifurcations and isolated singularities for robots and mechanisms. *IEEE Trans. Robotics Automat.*, **10**(1):1–10, 1994.
- [Lat93] J. C. Latombe, *Robot Motion Planning*. Kluwer, Boston, 1993.
- [Lié77] A. Liégeois, Automatic supervisory control of the configuration and behaviour of multibody mechanisms. *IEEE Trans. Systems, Man, Cybernet.*, **7**(6):868–871, 1977.
- [MLS94] R. M. Murray, Z. Li i S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton, 1994.
- [MT96] R. Muszyński i K. Tchoń, Normal forms of non-redundant singular robot kinematics: Three DOF worked examples. *J. Robotic Systems*, **13**(12):765–791, 1996.
- [MT98] R. Muszyński i K. Tchoń, A solution to the singular inverse kinematic problem for a planar manipulation robot mounted on a track. W: *1st IFAC Workshop on Space Robotics*, vol. 1, strony 146–151, Motreal, 1998.
- [Mus96] R. Muszyński, *Modele i algorytmy sterowania manipulatorów z osobliwościami kinematycznymi*. Rozprawa doktorska, Instytut Cybernetyki Technicznej, Politechnika Wrocławska, 1996.
- [Nak91] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*. Addison-Wesley, Reading, 1991.
- [Nen95] D. N. Nenchev, Tracking manipulator trajectories with ordinary singularities: A null space-based approach. *Int. J. Robotics Research*, **14**(4):399–404, 1995.
- [NH86] Y. Nakamura i H. Hanafusa, Inverse kinematic solutions with singularity robustness for robot manipulator control. *J. Dyn. Syst. Meas. Contr.*, **109**(2):163–171, 1986.
- [NTU98] D. N. Nenchev, Y. Tsumaki i M. Uchiyama, Singularity-consistent behavior of telerobots: Theory and experiments. *Int. J. Robotics Research*, **17**(2):138–152, 1998.
- [NU97] D. N. Nenchev i M. Uchiyama, Singularity-consistent path planning and motion control through instantaneous self-motion singularities of parallel-link manipulators. *J. Robotic Systems*, **14**(13):27–36, 1997.
- [OCS97] K. A. O’Neil, Y. C. Chen i J. Seng, Removing singularities of resolved motion rate control of mechanisms, including self-motion. *IEEE Trans. Robotics Automat.*, **13**(5):741–751, 1997.

- [PB94] F. C. Park i R. W. Brockett, Kinematic dexterity of robotic mechanisms. *Int. J. Robotics Research*, **13**(1):1–15, 1994.
- [Piw98] S. Piwko, Unikanie osobliwości w robotach o redundantnej kinematyce. Praca magisterska, Instytut Cybernetyki Technicznej, Politechnika Wroclawska, 1998.
- [PL92] D. K. Pai i M. C. Leu, Genericity and singularities of robot manipulators. *IEEE Trans. Robotics Automat.*, **8**(5):492–504, 1992.
- [RM94] R. G. Roberts i A. A. Maciejewski, Singularities, stable surfaces, and the repeatable behaviour of kinematically redundant manipulators. *Int. J. Robotics Research*, **13**(1):207–213, 1994.
- [SB96] A. Stokes i R. W. Brockett, Dynamics of kinematic chains. *Int. J. Robotics Research*, **15**(4):393–405, 1996.
- [SF88] M. Sampei i K. Furuta, Robot control in a neighbourhood of singular points. *J. Robotic Automat.*, **4**(3):303–309, 1988.
- [Sha90] T. Shamir, The singularities of redundant robot arm. *Int. J. Robotics Research*, **9**(1):113–121, 1990.
- [Sma76] S. Smale, A convergent process of price adjustment and global Newton method. *J. Math. Econ.*, **3**:107–120, 1976.
- [SOC97] J. Seng, K. A. O’Neil i Y. C. Chen, On the existence and the manipulability recovery rate of self-motion at manipulator singularities. *Int. J. Robotics Research*, **16**(2):171–184, 1997.
- [SS95] S. Shankar i A. Saraf, Singularities in mechanisms I. *Mech. Machine Theory*, **30**:1139–1148, 1995.
- [SS96] L. Sciavicco i B. Siciliano, *Modeling and Control of Robot Manipulators*. The McGraw-Hill, New York, 1996.
- [SV97] M. Spong i M. Vidyasagar, *Dynamika i sterowanie robotów*. WNT, Warszawa, 1997.
- [SY88] T. Shamir i Y. Yomdin, Repeatability of redundant manipulators. mathematical solution of the problem. *IEEE Trans. Autom. Contr.*, **33**:1004–1009, 1988.
- [T<sup>+</sup>97] Y. Tsumaki et al., Teleoperation based on adjoint Jacobian approach. *Contr. Systems*, **17**(1):53–62, 1997.
- [Tch90] K. Tchoń, Towards a differential topological classification of robot manipulators. W: *Robust Control of Linear Systems and Nonlinear Control*, strony 565–574. Birkhäuser, Boston, 1990.

- [Tch91] K. Tchoń, Differential topology of the inverse kinematic problem for redundant robot manipulators. *Int. J. Robotics Research*, **10**(5):492–504, 1991.
- [Tch95a] K. Tchoń, A normal form of singular kinematics of robot manipulators with smallest degeneracy. *IEEE Trans. Robotics Automat.*, **11**(3):401–404, 1995.
- [Tch95b] K. Tchoń, Normal forms of kinematic singularities of 3R robot manipulators. *Appl. Mathematics and Computer Science*, **5**(2):391–407, 1995.
- [Tch97] K. Tchoń, Singularity avoidance in robotic manipulators: A differential form approach. *Systems & Contr. Lett.*, **30**(6):165–176, 1997.
- [Tch98] K. Tchoń, Quadratic normal forms of redundant robot kinematics with application to singularity avoidance. *IEEE Trans. Robotics Automat.*, **14**(5):834–837, 1998.
- [Tch99] K. Tchoń, Singularities of the Euler wrist. *Mech. Machine Theory*, 1999. W druku.
- [TD93] K. Tchoń i I. Dulęba, On inverting singular kinematics and geodesic trajectory generation for robot manipulators. *J. Intelligent Robotic Systems*, **8**:325–359, 1993.
- [TM95] K. Tchoń i A. Matuszok, On avoiding singularities in redundant robot kinematics. *Robotica*, **13**:599–606, 1995.
- [TM97] K. Tchoń i R. Muszyński, Singularities of non-redundant robot kinematics. *Int. J. Robotics Research*, **16**(1):60–76, 1997.
- [TM98] K. Tchoń i R. Muszyński, Singular inverse kinematic problem for robotic manipulators: A normal form approach. *IEEE Trans. Robotics Automat.*, **14**(1):93–104, 1998.
- [TMZ97] K. Tchoń, R. Muszyński i C. Zieliński, Kinematic singularities of the IRp-6 manipulator mounted on a track. W: *Proc. MMAR Symposium*, vol. **3**, strony 913–920, Międzyzdroje, 1997.
- [Wam86] C. W. Wampler II, Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. *IEEE Trans. Systems, Man, Cybernet.*, **16**:93–101, 1986.
- [Yos84] T. Yoshikawa, Analysis and control of robot manipulators with redundancy. W: M. Brady i R. Paul, (red.), *Robotics Research*, strony 735–747. MIT Press, Cambridge, 1984.
- [Yos91] T. Yoshikawa, Manipulability of robotic mechanisms. *Int. J. Robotics Research*, **4**(2):3–9, 1991.



## Rozdział 4

# Kinematyka odwrotna robotów mobilnych

W podrozdziale 2.4 pokazaliśmy, że modelem kinematyki robota mobilnego jest pewien układ sterowania. Wyznaczając trajektorie tego układu dla danego stanu początkowego i różnych sterowań dopuszczalnych potrafimy określić zbiór stanów osiągalnych układu w każdej chwili czasu. Przez potraktowanie kinematyki robota mobilnego jako odwzorowania osiągalności układu sterowania skonstruujemy odpowiednik odwzorowania przestrzeni przegubowej w przestrzeń zadaniową, które stanowi model kinematyki manipulatora. Ta odpowiedniość pozwoli nam na zastosowanie do rozwiązania odwrotnego zadania kinematyki robota mobilnego algorytmu Newtona, zdefiniowanego w przestrzeni sterowań. Jak zobaczymy w III części książki, postać modelu kinematyki robota mobilnego pozwala także na zastosowanie innych, bardziej specyficznych algorytmów kinematyki odwrotnej. Wszelako podejście oparte na algorytmie Newtona zasługuje na wyróżnienie ze względu na jego adekwatność do obu klas robotów: manipulacyjnych i mobilnych.

### 4.1 Metoda Newtona

Niech będzie dana kinematyka robota mobilnego w formie układu sterowania

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{q})u_i, \quad (4.1)$$

który przy ustalonym stanie początkowym  $\mathbf{q}_0 \in \mathbb{R}^n$  i horyzoncie czasowym  $T > 0$  definiuje odwzorowanie  $\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$  (2.147), oraz niech  $\mathbf{q}_d \in \mathbb{R}^n$  oznacza stan docelowy. Zadanie wyboru sterowania  $\mathbf{u}(\cdot) \in \mathbb{L}_m^2[0, T]$ , takiego że

$$\boldsymbol{\varphi}_T(\mathbf{q}_0, \mathbf{u}(\cdot)) = \mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot)) = \mathbf{q}_d,$$

nazywamy *odwrotnym zadaniem kinematyki* lub *zadaniem planowania ruchu* robota mobilnego. Naszym celem będzie przedstawienie *metody Newtona* rozwiązywania odwrotnego zadania kinematyki robota mobilnego, korzystającej z pojęcia jakobianu i analogicznej do metod jakobianowych kinematyki odwrotnej manipulatorów. Aspekty obliczeniowe metody Newtona, a także inne algorytmy planowania ruchu robota mobilnego zostaną omówione w III części książki.

Niech  $D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$  oznacza jakobian analityczny (2.147). Przy ustalonym  $\mathbf{u}(\cdot)$  i dla zadanego wektora  $\mathbf{w} \in \mathbb{R}^n$  (kierunku ruchu w przestrzeni zadaniowej robota mobilnego) rozważmy równanie liniowe

$$D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))\mathbf{v}(\cdot) = \mathbf{w},$$

którego rozwiązanie prowadzi do wyznaczenia sterowania  $\mathbf{v}(\cdot)$ . Podobnie jak to zrobiliśmy w przypadku skończonego wymiarowego, wyliczymy obecnie pseudoodwrotność jakobianu  $D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$ . Pseudoodwrotność ta stanowi rozwiązanie następującego izoperymetrycznego zadania wariacyjnego:

Znaleźć minimum funkcjonału

$$\mathcal{J}(\mathbf{v}(\cdot)) = \frac{1}{2} \|\mathbf{v}(\cdot)\|^2 = \frac{1}{2} \int_0^T \mathbf{v}^T(s) \mathbf{v}(s) ds \quad (4.2)$$

przy ograniczeniu równościowym

$$D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))\mathbf{v}(\cdot) = \int_0^T \boldsymbol{\Phi}(T, s) \mathbf{B}(s) \mathbf{v}(s) ds = \mathbf{w}.$$

Standardowa metoda rozwiązania tego zadania wymaga wprowadzenia rozszerzonej funkcji podcałkowej funkcjonału (4.2)

$$F(\mathbf{v}, \boldsymbol{\lambda}, t) = \frac{1}{2} \mathbf{v}^T \mathbf{v} + \boldsymbol{\lambda}^T \boldsymbol{\Phi}(T, t) \mathbf{B}(t) \mathbf{v}$$

zawierającej wektor mnożników Lagrange'a  $\boldsymbol{\lambda} \in \mathbb{R}^n$  i rozwiązania odpowiednich równań Eulera-Lagrange'a. Po wyeliminowaniu wektora  $\boldsymbol{\lambda}$  otrzymujemy następujące rozwiązanie

$$\mathbf{v}(t) = \mathbf{B}^\top(t)\boldsymbol{\Phi}^\top(T, t)\mathbf{M}_{\mathbf{q}_0, T}^{-1}(\mathbf{u}(\cdot))\mathbf{w}. \quad (4.3)$$

Macierz

$$\mathbf{M}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot)) = \int_0^T \boldsymbol{\Phi}(T, s)\mathbf{B}(s)\mathbf{B}^\top(s)\boldsymbol{\Phi}^\top(T, s) ds, \quad (4.4)$$

której odwrotność występuje w zależności (4.3) jest znana jako *macierz Grama* liniowego, zależnego od czasu układu sterowania\*

$$\dot{\boldsymbol{\xi}} = \mathbf{A}(t)\boldsymbol{\xi} + \mathbf{B}(t)\mathbf{v}, \quad (4.5)$$

stowarzyszonego z układem podstawowym (4.1) i określonego wzdłuż pary (*sterowanie, trajektoria*) =  $(\mathbf{u}(t), \mathbf{q}(t))$ . Przypomnijmy, że macierze układu (4.5) zostały zdefiniowane w następujący sposób

$$\mathbf{A}(t) = \frac{\partial(\mathbf{G}(\mathbf{q}(t))\mathbf{u}(t))}{\partial \mathbf{q}}, \quad \mathbf{B}(t) = \mathbf{G}(\mathbf{q}(t)). \quad (4.6)$$

Nieosobliwość (dodatnia określoność) macierzy Grama jest warunkiem koniecznym i wystarczającym nieosobliwości sterowania  $\mathbf{u}(\cdot)^\dagger$ , co stanowi zarazem warunek konieczny i wystarczający sterowalności układu (4.5). Zauważmy, że przy nieosobliwym sterowaniu formuła (4.3) definiuje (prawostronną) *pseudoodwrotność* jacobianu analitycznego robota mobilnego

$$\begin{aligned} & (D \mathbf{k}_{\mathbf{q}_0, T})^\#(\mathbf{u}(\cdot)) : \mathbb{R}^n \longrightarrow \mathbb{L}_m^2[0, T], \\ & (D \mathbf{k}_{\mathbf{q}_0, T})^\#(\mathbf{u}(\cdot))(\mathbf{w})(t) = \mathbf{B}^\top(t)\boldsymbol{\Phi}^\top(T, t)\mathbf{M}_{\mathbf{q}_0, T}^{-1}(\mathbf{u}(\cdot))\mathbf{w}. \end{aligned} \quad (4.7)$$

Nietrudno bowiem pokazać, że

$$D \mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))(D \mathbf{k}_{\mathbf{q}_0, T})^\#(\mathbf{u}(\cdot))(\mathbf{w}) = \mathbf{w}.$$

Po dokonaniu niezbędnych przygotowań, możemy teraz przedstawić *algorytm kinematyki odwrotnej* robota mobilnego, będący odpowiednikiem algorytmu (3.29) kinematyki odwrotnej manipulatora (algorytmu Newtona).

\*Porównaj z (2.150).

†Zobacz podrozdział 2.4.

W myśl algorytmu Newtona, sterowanie  $\mathbf{u}(\cdot)$  powinno być elementem 1-parametrowej rodziny sterowań  $\{\mathbf{u}_\tau(\cdot) | \tau \in \mathbb{R}\}$  w przestrzeni  $\mathbb{L}_m^2[0, T]$  spełniającym równanie

$$\frac{d\mathbf{u}_\tau(\cdot)}{d\tau} = -\alpha(D\mathbf{k}_{\mathbf{q}_0, T})^\#(\mathbf{u}_\tau(\cdot))(\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}_\tau(\cdot)) - \mathbf{q}_d). \quad (4.8)$$

W przedstawionym algorytmie  $(D\mathbf{k}_{\mathbf{q}_0, T})^\#(\mathbf{u}_\tau(\cdot))$  oznacza pseudoodwrotność (4.7) jacobianu analitycznego, zaś  $\alpha > 0$  jest parametrem. Przy założeniu, że jacobian robota mobilnego  $D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}_\tau(\cdot))$  jest pełnego rzędu wzdłuż krzywej  $\mathbf{u}_\tau(\cdot)$ , granica ciągu sterowań  $\lim_{\tau \rightarrow +\infty} \mathbf{u}_\tau(\cdot) = \mathbf{u}(\cdot)$  jest rozwiązaniem zadania odwrotnego  $\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot)) = \mathbf{q}_d$ . Korzystając z zależności (4.4) i (4.7) możemy zapisać algorytm (4.8) w chwili  $t$  jako

$$\begin{aligned} \frac{d\mathbf{u}_\tau(t)}{d\tau} &= -\alpha \mathbf{B}_\tau^\top(t) \Phi_\tau^\top(T, t) \\ &\left( \int_0^T \Phi_\tau(T, s) \mathbf{B}_\tau(s) \mathbf{B}_\tau^\top(s) \Phi_\tau^\top(T, s) ds \right)^{-1} (\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot)_\tau) - \mathbf{q}_d), \end{aligned} \quad (4.9)$$

gdzie macierz fundamentalna  $\Phi_\tau(t, s)$  została wyliczona z równania

$$\frac{d}{dt} \Phi_\tau(t, s) = \mathbf{A}_\tau(t) \Phi_\tau(t, s), \quad \Phi_\tau(s, s) = \mathbb{I}_n,$$

przy założeniu, że macierze  $\mathbf{A}_\tau(t)$ ,  $\mathbf{B}_\tau(t)$  zależą od aktualnej pary (sterowanie, trajektoria)  $= (\mathbf{u}_\tau(t), \mathbf{q}_\tau(t))$ , to jest

$$\begin{aligned} \dot{\mathbf{q}}_\tau(t) &= \mathbf{G}(\mathbf{q}_\tau(t)) \mathbf{u}_\tau(t), \\ \mathbf{A}_\tau(t) &= \frac{\partial(\mathbf{G}(\mathbf{q}_\tau(t)) \mathbf{u}_\tau(t))}{\partial \mathbf{q}}, \quad \mathbf{B}_\tau(t) = \mathbf{G}(\mathbf{q}_\tau(t)). \end{aligned} \quad (4.10)$$

Jak widać, algorytm (4.8) kinematyki odwrotnej robota mobilnego wymaga nieosobliwości sterowania  $\mathbf{u}(\cdot)$ , która jest równoważna dodatniej określoności macierzy Grama. Warunki konieczne i wystarczające nieosobliwości sterowania są następujące.

**Twierdzenie 4.1.1** *Macierz Grama  $\mathbf{M}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$  jest dodatnio określona wtedy i tylko wtedy, gdy nie istnieje wektor  $\mathbf{p} \in \mathbb{R}^n$ ,  $\mathbf{p} \neq \mathbf{0}$ , taki że*

$$\mathbf{p}^\top \Phi(T, t) \mathbf{B}_i(t) = 0 \quad (4.11)$$

dla każdego  $t \in [0, T]$  oraz  $i = 1, 2, \dots, m$ . W równaniu (4.11)  $\Phi(t, s)$  oznacza macierz fundamentalną układu (4.5), natomiast  $\mathbf{B}_i(t)$  jest  $i$ -tą

kolumną macierzy  $\mathbf{B}(t)$  zdefiniowanej formułą (4.6). Warunek (4.11) jest równoważny stwierdzeniu, że nie istnieje różne od zera rozwiązanie  $\mathbf{p}(t)$  równania dotychczasowego

$$\dot{\mathbf{p}} = - \left( \frac{\partial(\mathbf{G}(\mathbf{q}(t))\mathbf{u}(t))}{\partial \mathbf{q}} \right)^T \mathbf{p}, \quad (4.12)$$

takie że

$$\mathbf{p}^T \mathbf{g}_i(\mathbf{q}) = 0 \quad (4.13)$$

dla  $t \in [0, T]$  oraz  $i = 1, 2, \dots, m$ .

Zauważmy, że druga część powyższego twierdzenia orzeka, że macierz Grama jest dodatnio określona wtedy i tylko wtedy, gdy ekstremala  $(\mathbf{u}(t), \mathbf{q}(t), \mathbf{p}(t))$  nie jest ekstremalą osobliwą układu (4.1)<sup>‡</sup>, a zatem stanowi uzasadnienie używanej przez nas wcześniej formuły (2.153).

Ze względu na konieczność obliczania macierzy fundamentalnej  $\Phi(t, s)$ , wyznaczenie macierzy Grama (4.4), a więc badanie osobliwości sterowania, jest w ogólnym przypadku możliwe wyłącznie na drodze numerycznej. Jednak, jeżeli wiadomo, że sterowanie  $\mathbf{u}(\cdot)$  jest gładką lub analityczną funkcją czasu, to korzystając z tego, że pola wektorowe  $\mathbf{g}_i(\mathbf{q})$  układu (4.1) są zawsze gładkie lub analityczne, można wyprowadzić z warunku (4.13) następujące warunki nieosobliwości sterowania.

**Twierdzenie 4.1.2** *Załóżmy, że zarówno układ (4.1), jak i sterowania  $\mathbf{u}(\cdot)$  są gładkie. Dla układu stowarzyszonego (4.5) zdefiniujmy ciąg macierzy*

$$\mathbf{B}_0(t) = \mathbf{B}(t), \quad \mathbf{B}_{i+1}(t) = \mathbf{A}(t)\mathbf{B}_i(t) - \frac{d}{dt}\mathbf{B}_i(t), \quad i \geq 0.$$

*Wówczas, jeżeli istnieje takie naturalne  $k$ , że w pewnej chwili  $s \in [0, T]$*

$$\text{rank} [\mathbf{B}_0(s), \mathbf{B}_1(s), \dots, \mathbf{B}_k(s)] = n, \quad (4.14)$$

*to sterowanie  $\mathbf{u}(\cdot)$  jest nieosobliwe. Jeżeli układ (4.1) i sterowania są analityczne, warunek (4.14) jest warunkiem koniecznym i wystarczającym nieosobliwości sterowania  $\mathbf{u}(\cdot)$ .*

<sup>‡</sup>Zobacz dodatek A.4.

## 4.2 Elipsoida mobilności

Przez analogię do koncepcji manipulowalności manipulatorów, wykorzystamy teraz macierz Grama  $\mathbf{M}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$  do wprowadzenia dla robotów mobilnych odpowiednika pojęcia manipulowalności, zwanego *mobilnością* robota. Przede wszystkim zauważmy, że na mocy definicji (2.148) jacobian  $D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$  przy ustalonym sterowaniu  $\mathbf{u}(\cdot) \in \mathbb{L}_m^2[0, T]$  stanowi przekształcenie prędkości zmian (wariacji) sterowań w prędkość ruchu robota mobilnego,

$$D\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))\mathbf{v}(\cdot) = \mathbf{w}.$$

Ponieważ przestrzeń wariacji sterowań ( $= \mathbb{L}_m^2[0, T]$ ) jest unormowana, możemy zdefiniować w niej sferę jednostkową

$$S = \left\{ \mathbf{v}(\cdot) \in \mathbb{L}_m^2[0, T] \mid \|\mathbf{v}(\cdot)\|^2 = \int_0^T \mathbf{v}^T(t)\mathbf{v}(t) dt = 1 \right\}. \quad (4.15)$$

Zbadajmy obraz tej sfery w przestrzeni prędkości ruchu robota mobilnego. Ponieważ  $\mathbf{v}(\cdot) = (D\mathbf{k}_{\mathbf{q}_0, T})^\#(\mathbf{u}(\cdot))\mathbf{w}$ , obrazem sfery (4.15) jest elipsoida, którą będziemy nazywać *elipsoidą mobilności*

$$\begin{aligned} E_m^{\mathbf{q}_0, T}(\mathbf{u}(\cdot)) &= \left\{ \mathbf{w} \mid ((D\mathbf{k}_{\mathbf{q}_0, T})^\#(\mathbf{u}(\cdot))\mathbf{w})^T (D\mathbf{k}_{\mathbf{q}_0, T})^\#(\mathbf{u}(\cdot))\mathbf{w} = \right. \\ &= \left. \mathbf{w}^T \mathbf{M}_{\mathbf{q}_0, T}^{-1}(\mathbf{u}(\cdot))\mathbf{w} = 1 \right\}. \quad (4.16) \end{aligned}$$

Długość najkrótszej i najdłuższej półosi elipsoidy mobilności wyznaczają, odpowiednio, pierwiastki kwadratowe z najmniejszej i największej wartości własnej macierzy  $\mathbf{M}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$ . Macierz  $\mathbf{M}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$  będziemy nazywać *macierzą mobilności* sterowania  $\mathbf{u}(\cdot)$  przy ustalonych  $\mathbf{q}_0$  oraz  $T$ . Miarę objętości elipsoidy mobilności  $E_m^{\mathbf{q}_0, T}(\mathbf{u}(\cdot))$

$$m_{\mathbf{q}_0, T}(\mathbf{u}(\cdot)) = \sqrt{\det \mathbf{M}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot))} \quad (4.17)$$

nazwiemy *mobilnością* robota mobilnego przy sterowaniu  $\mathbf{u}(\cdot)$ . Jak łatwo zauważyć, przy sterowaniach osobliwych mobilność robota spada do zera.

## 4.3 Komentarze i uwagi bibliograficzne

Przedstawienie kinematyki robota mobilnego w formie nieliniowego układu sterowania (4.1) prowadzi w naturalny sposób do określenia kinematyki jako

odwzorowania osiągalności tego układu i sformułowania odwrotnego zadania kinematyki jako zadania poszukiwania sterowań zapewniających osiągnięcie przez układ sterowania zadanego stanu docelowego. Tak postawione zadanie odwrotne bywa często nazywane zadaniem planowania ruchu robota mobilnego [Lat93]. W niniejszym rozdziale zwróciliśmy uwagę tylko na dwa aspekty zadania planowania ruchu, mianowicie na możliwość zastosowania do rozwiązania tego zadania algorytmu Newtona w nieskończenie wymiarowej przestrzeni sterowań układu (4.1) oraz na możliwość zdefiniowania elipsoidy mobilności i zaproponowania kryterium mobilności robota mobilnego w sposób analogiczny do miary manipulowalności manipulatora. Inne aspekty problematyki planowaniu ruchu, sterowania i nawigacji robotów mobilnych będą omawiane w III części książki. Zadanie planowania ruchu robotów mobilnych posiada bardzo bogatą literaturę. Pierwsza praca na ten temat, poświęcona zagadnieniu optymalnego układania torów kolejowych, została napisana przez Markowa ponad sto lat temu [Mar87]. Do pionierskich prac współczesnych należą artykuły Dubinsa [Dub57] oraz Reedsa i Sheppa [RS90]. Rozwój tematyki planowania, jaki nastąpił w latach 90-tych, wynikał z zainteresowania planowaniem ruchu zarówno robotyków [LC93, MLS94, CSB96, Lau98], jak i matematyków, którzy traktowali zadanie planowania jako pewne zadanie geometrii subriemannowskiej [Str83, VG94, Mon95, AS95, BR96]. Na postęp w dziedzinie metod planowania ruchu miał znaczny wpływ rozwój geometrycznej teorii sterowania [Isi89, NS90], której wyniki znalazły bezpośrednie zastosowanie do układów typu (4.1) [Her78, Her80, Bro81, Bro82, Sus82, Sus90, F<sup>+</sup>95, JkR98]. Wyprowadzenie postaci macierzy Grama (4.4) układu stowarzyszonego z (4.1), oraz pseudoodwrotności jacobianu analitycznego (4.7) można znaleźć w monografii Sontaga [Son90]. Stamtąd też pochodzi twierdzenie 4.1.1 i warunek sterowalności przedstawiony w twierdzeniu 4.1.2. Dowód tego warunku, wykorzystujący twierdzenie Chow, zawiera rozprawa doktorska Lobry'ego [Lob72]. Algorytm Newtona (4.8) pochodzi od Wena [DW94, PW96], patrz także [Son98]. Na znaczenie sterowań osobliwych dla klasyfikacji układów sterowania zwrócił uwagę Bonnard [Bon91, Bon98]. Odpowiadające tym sterowaniom tzw. krzywe abnormalne odgrywają ważną rolę przy geometrycznej klasyfikacji dystrybucji [Mon95, Zhi95]. Nowy niezmiennik tzw. struktur Goursata, których modelem jest robot mobilny złożony z traktora ciągnącego przyczepy, zwany typem osobliwości, i jego związek z wektorem wzrostu dystrybucji oraz krzywymi abnormalnymi został wykryty w pracy [PLR99]. Pojęcie elipsoidy

mobilności i miary mobilności (4.17) zostały wprowadzone po raz pierwszy w tej książce. Naturalne rozszerzenie zadania planowania ruchu do układów robotycznych złożonych z platformy mobilnej i zamontowanego na niej manipulatora (tzw. manipulatorów mobilnych) jest aktualnie przedmiotem naszych badań.

## Literatura

- [AS95] A. A. Agrachev i A. V. Sarychev, Strong minimality of abnormal geodesics for 2-distributions. *J. Dyn. Contr. Syst.*, 1(2):139–176, 1995.
- [BMS82] R. W. Brockett, R. S. Millman i H. J. Sussmann, (red.), *Differential Geometric Control Theory*. Birkhäuser, Boston, 1982.
- [Bon91] B. Bonnard, Feedback equivalence for nonlinear control systems and the time optimal control problem. *SIAM J. Contr. Opt.*, 29:1300–1321, 1991.
- [Bon98] B. Bonnard, Singular trajectories, feedback equivalence and the time optimal control problem. W: Jakubczyk i Respondek [JkR98], strony 79–110.
- [BR96] A. Bellaïche i I. J. Risler, (red.), *Sub-Riemannian Geometry*. Birkhäuser, Boston, 1996.
- [Bro81] R. W. Brockett, Control theory and singular Riemannian geometry. W: P. Hinton i G. Young, (red.), *New Directions in Applied Mathematics*, strony 11–27. Springer-Verlag, New York, 1981.
- [Bro82] R. W. Brockett, Asymptotic stability and feedback stabilization. W: Brockett et al. [BMS82], strony 181–191.
- [CSB96] C. Canudas de Wit, B. Siciliano i G. Bastin, *Theory of Robot Control*. Springer-Verlag, New York, 1996.
- [Dub57] L. E. Dubins, On curves of minimal length with a constrained on average curvature and with prescribed initial and terminal positions and tangents. *American J. Math.*, strony 497–516, 1957.
- [DW94] A. W. Divelbiss i J. T. Wen, Nonholonomic path planning with inequality constraints. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 52–57, San Diego, 1994.
- [F<sup>+</sup>95] M. Fliess et al., Flatness and defect on non-linear systems: Introductory theory and examples. *Int. J. Control*, 61:1327–1361, 1995.



- [Her78] H. Hermes, Lie algebras of vector fields and local approximation of attainable sets. *SIAM J. Contr. Opt.*, **16**(6):715–728, 1978.
- [Her80] H. Hermes, On the synthesis of a stabilizing feedback control via Lie algebraic method. *SIAM J. Contr. Opt.*, **18**(6):352–361, 1980.
- [Isi89] A. Isidori, *Nonlinear Control Systems*. Springer-Verlag, Berlin, 1989.
- [JkR98] B. Jakubczyk i W. Respondek, (red.), *Geometry of Feedback and Optimal Control*. M. Dekker, New York, 1998.
- [Lat93] J. C. Latombe, *Robot Motion Planning*. Kluwer, Boston, 1993.
- [Lau98] J. P. Laumond, (red.), *Robot Motion Planning and Control*. Lecture Notes in Control and Information Sciences. Springer-Verlag, New York, 1998.
- [LC93] Z. Li i J. Canny, (red.), *Nonholonomic Motion Planning*. Kluwer, New York, 1993.
- [Lob72] C. Lobry, *Queleques Aspect Qualitatifs de la Théorie de la Commande*. Rozprawa doktorska, Université de Grenoble, Grenoble, 1972.
- [Mar87] A. A. Markov, Some examples of the solution of a special kind of problem on greatest and least quantities. *Soobshch. Kharkovsk. Mat. Obshch.*, **1**:250–276, 1887. w j. ros.
- [MLS94] R. M. Murray, Z. Li i S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton, 1994.
- [Mon95] R. Montgomery, A survey of singular curves in sub-Riemannian geometry. *J. Dyn. Contr. Syst.*, **1**(1):49–90, 1995.
- [NS90] H. Nijmeijer i A. J. van der Schaft, *Nonlinear Dynamic Control Systems*. Springer-Verlag, New York, 1990.
- [PLR99] W. Pasillas-Lépine i W. Respondek, On the geometry of Goursat structures. Preprint, 1999.
- [PW96] D. O. Popa i J. T. Wen, Nonholonomic path-planning with obstacle avoidance: A path-space approach. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 4, strony 2662–2667, Minneapolis, 1996.
- [RS90] J. A. Reeds i R. A. Shepp, Optimal paths for a car that goes both forward and backwards. *Pacific Journ. Math.*, **145**(2):367–393, 1990.
- [Son90] E. D. Sontag, *Mathematical Control Theory*. Springer-Verlag, New York, 1990.
- [Son98] E. D. Sontag, A general approach to path planning for systems without drift. W: J. Bailleul, S. S. Sastry i H. J. Sussmann, (red.), *Essays on Mathematical Robotics*, strony 151–168. Springer-Verlag, New York, 1998.

- [Str83] R. S. Strichartz, Sub-Riemannian geometry. *J. Diff. Geom.*, **24**:221–263, 1983.
- [Sus82] H. J. Sussmann, Lie brackets, real analyticity and geometric control. W: Brockett et al. [BMS82], strony 1–116.
- [Sus90] H. J. Sussmann, (red.), *Differential Geometric Control Theory*. M. Dekker, New York, 1990.
- [VG94] A. M. Vershik i V. Ya. Gershkovich, Nonholonomic dynamical systems, geometry of distribution and variational problems. W: V. I. Arnold, (red.), *Dynamical Systems VII*, strony 1–81. Springer-Verlag, Berlin, 1994.
- [Zhi95] M. Zhitomirskii, Rigid and abnormal line subdistributions of 2-distributions. *J. Dyn. Contr. Syst.*, **1**(2):253–294, 1995.

Część II

**Algorytmy sterowania  
manipulatorów**



## Rozdział 5

# Algorytmy sterowania manipulatorów sztywnych w przestrzeni zadaniowej

Rozdział ten jest poświęcony zagadnieniom sterowania manipulatorów sztywnych, a więc układów robotycznych o nieruchomej podstawie (bazie) zbudowanych z łańcucha ramion będących ciałami sztywnymi, połączonych sztywnymi przegubami. Manipulator może być częścią robota manipulacyjnego lub być umieszczony na platformie mobilnej\*. Zgodnie z klasycznym paradygmatem automatyki, najwcześniejsze algorytmy sterowania używane w robotach traktowały dynamikę manipulatora jako zbiór nie sprzężonych ze sobą liniowych obiektów sterowania rzędu drugiego, opisujących pojedyncze stopnie swobody. Do każdego z tych obiektów stosowano regulator typu PD lub PID. To tradycyjne podejście dominuje nadal w robotach przemysłowych, nie jest ono jednak w stanie zagwarantować odpowiedniej jakości sterowania wymaganej we współczesnych zastosowaniach robotów. Wiadomo bowiem, że podczas realizacji ruchów o dużym zakresie zmian położenia i prędkości, istotną rolę zaczynają odgrywać nieliniowości dynamiki oraz sprzężenia dynamiczne między ogniwami. Wysokie wymagania dotyczące stabilności i dokładności stawiane obecnie układom sterowania manipulatorów mogą być spełnione wtedy, gdy algorytmy sterowania opierają się na kompletnych, nieliniowych równaniach dynamiki manipulatora. Gdy model dynamiki manipulatora nie jest w pełni znany, używane są algorytmy sterowania adaptacyjnego lub odpornego. Algorytmy sterowania

---

\*Taki układ robotyczny nazywa się manipulatorem mobilnym.

wykorzystujące model dynamiki manipulatora zaczynają obecnie być stosowane przez czołowych producentów w robotach przemysłowych.

Zadania sterowania manipulatora można sklasyfikować w zależności od pożądanego zachowania efektora. Najczęstszym celem sterowania jest przeprowadzenie efektora od określonego punktu początkowego do punktu końcowego przestrzeni zadaniowej. W nowoczesnych robotach pojawia się jednak potrzeba zapewnienia ruchu efektora wzdłuż zadanej, zależnej od czasu trajektorii. Taki ruch jest niezbędny w przypadku, gdy zadaniem robota jest spawanie, malowanie, skrawanie, cięcie lub operacje montażowe, a w niedalekiej przyszłości precyzyjne operacje chirurgiczne, itp.

W naszych rozważaniach skoncentrujemy się na algorytmach sterowania zapewniających śledzenie zadanej trajektorii (zadaniowej lub przegubowej). Przyjmijmy założenie o regularności trajektorii zadanej, a mianowicie założymy, że zadana trajektoria jest gładka i ograniczona wraz z dwiema (dla manipulatora sztywnego) lub czterema (w przypadku manipulatora o elastycznych przegubach) pierwszymi pochodnymi względem czasu.

W dalszej części tego rozdziału przedstawimy zagadnienie sterowania w przestrzeni zadaniowej manipulatora, a więc w przypadku, gdy została zadana trajektoria efektora. W rozdziale 6 opiszemy algorytmy sterowania manipulatorów sztywnych, dla których jest możliwe przekształcenie trajektorii określonej w przestrzeni zadaniowej w trajektorię zadaną w przestrzeni przegubowej. Osobnym zadaniem jest uzyskanie algorytmów sterowania dla manipulatorów elastycznych. W praktyce mogą wystąpić dwa rodzaje zjawisk elastycznych, a mianowicie elastyczność przegubów lub elastyczność ramion manipulatora. Uwzględnienie efektów elastycznych powoduje rozszerzenie modelu matematycznego opisującego zachowanie manipulatora (modelu dynamiki) i wymaga zastosowania specyficznych algorytmów sterowania, innych niż w przypadku manipulatorów sztywnych. Algorytmy sterowania manipulatorów z elastycznymi przegubami zostaną omówione w rozdziale 7.

## 5.1 Dynamika układu robotycznego

Dla układu robotycznego z uniwersum fazowym  $\mathbb{R}^N \times \mathbb{R}^N$ , nie podlegającemu ograniczeniom konfiguracyjnym ani fazowym, a więc o liczbie stopni swobody  $n = N$ , definiujemy gładką (analityczną) funkcję Lagrange'a  $L(\mathbf{q}, \dot{\mathbf{q}})$ , zwaną *lagranżianem*, rozumianą jako różnica *energii kinetycznej* i *potencjalnej* układu,  $L(\mathbf{q}, \dot{\mathbf{q}}) = K(\mathbf{q}, \dot{\mathbf{q}}) - V(\mathbf{q})$ . Na mocy *Zasady Najmniejszego*

Działania Hamiltona, równania dynamiki układu robotycznego przyjmują postać równań Eulera-Lagrange'a\*

$$\frac{d}{dt} \frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} - \frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} = \mathbf{F}, \quad (5.1)$$

gdzie wektor  $\mathbf{F}$  symbolizuje uogólnione siły niepotencjalne działające na układ (tarcie, opory ruchu, siły więzów, oddziaływania sterujące itp.). Układ (5.1) jest układem równań różniczkowych zwyczajnych drugiego rzędu,

$$\frac{\partial^2 L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}^2} \ddot{\mathbf{q}} + \frac{\partial^2 L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q} \partial \dot{\mathbf{q}}} \dot{\mathbf{q}} - \frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} = \mathbf{F}. \quad (5.2)$$

Zważywszy, że energia kinetyczna układu robotycznego ma postać formy kwadratowej prędkości uogólnionej z symetryczną i dodatnio określoną macierzą formy zależną od współrzędnych uogólnionych, tzn.

$$K(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{Q}(\mathbf{q}) \dot{\mathbf{q}},$$

uzyskujemy na podstawie (5.2) następującą ogólną postać równań dynamiki układu robotycznego

$$\mathbf{Q}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) = \mathbf{F}. \quad (5.3)$$

W powyższym równaniu macierz formy energii kinetycznej  $\mathbf{Q}(\mathbf{q})$  pełni rolę macierzy inercji układu, wektor  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}$  opisuje wpływ sił (momentów sił) Coriolisa i odśrodkowych, a wektor  $\mathbf{D}(\mathbf{q}) = \frac{\partial V(\mathbf{q})}{\partial \mathbf{q}}$  jest wektorem sił grawitacji. Elementy macierzy sił Coriolisa i odśrodkowych  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$  można wyrazić przez elementy macierzy inercji w następujący sposób

$$C_{ij}(\mathbf{q}, \dot{\mathbf{q}}) = \sum_{k=1}^n c_{kj}^i(\mathbf{q}) \dot{q}_k, \quad (5.4)$$

gdzie współczynniki

$$c_{kj}^i(\mathbf{q}) = \frac{1}{2} \left( \frac{\partial Q_{ij}(\mathbf{q})}{\partial q_k} + \frac{\partial Q_{ik}(\mathbf{q})}{\partial q_j} - \frac{\partial Q_{jk}(\mathbf{q})}{\partial q_i} \right) \quad (5.5)$$

nazywają się *symbolami Christoffela* I rodzaju macierzy inercji.

---

\*Zobacz dodatek A.6.

Bezpośrednio z definicji wynikają następujące własności modelu dynamiki (5.3):

- macierz inercji  $\mathbf{Q}(\mathbf{q})$  jest symetryczna i dodatnio określona (a więc nieosobliwa),
- pochodna macierzy inercji wzdłuż trajektorii przegubowej jest wyznaczona przez macierz sił Coriolisa i odśrodkowych

$$\frac{d}{dt}\mathbf{Q}(\mathbf{q}) = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{C}^T(\mathbf{q}, \dot{\mathbf{q}}). \quad (5.6)$$

Zakładając, że na  $i$ -tą współrzędną uogólnioną układu działa sterowanie  $u_i$ , oraz że wszelkie inne niepotencjalne siły uogólnione (np. siły tarcia) można opisać pewną funkcją  $-\zeta(\mathbf{q}, \dot{\mathbf{q}}, t)$ , model dynamiki (5.3) układu robotycznego bez ograniczeń przyjmuje postać

$$\mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) + \zeta(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{u}, \quad (5.7)$$

gdzie  $\mathbf{u} = (u_1, \dots, u_n)^T$ . W następnym podrozdziale wyznaczymy poszczególne składniki modelu dynamiki układu robotycznego, jakim jest manipulator o sztywnych ramionach i sztywnych przegubach, zwany w skrócie *manipulatorem sztywnym*.

## 5.2 Dynamika manipulatora sztywnego

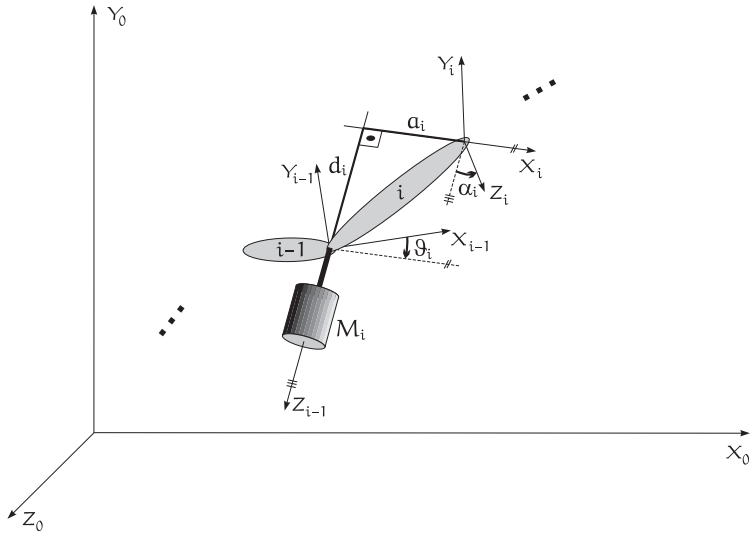
Energia kinetyczna i potencjalna manipulatora sztywnego o  $n$  stopniach swobody składa się z energii kinetycznej i energii potencjalnej każdego z ramion manipulatora wraz z jego układem napędowym. Dwa kolejne ramiona i układ napędowy przedstawia schematycznie rysunek 5.1.

Rozważmy najpierw energię kinetyczną elementu masy  $dm$  ramienia o numerze  $i$ , którego położenie względem układu  $X_i Y_i Z_i$  związanego z tym ramieniem jest określone przy pomocy współrzędnych jednorodnych  $\mathbf{r}_i = (x_i, y_i, z_i, 1)^T$ . Zakładając, że transformacja kinematyczna

$$\mathbf{A}_0^i(\mathbf{q}^i) = \prod_{k=1}^i \mathbf{A}_{k-1}^k(q_k),$$

gdzie  $\mathbf{q}^i = (q_1, \dots, q_i)^T$ , układu podstawowego  $X_0 Y_0 Z_0$  w układ  $X_i Y_i Z_i$  jest znana, obliczymy współrzędne elementu  $dm$  w układzie podstawowym jako  $\mathbf{A}_0^i(\mathbf{q}^i) \mathbf{r}_i$ . Prędkość elementu  $dm$  względem układu podstawowego





Rysunek 5.1 Ogniwko manipulatora wraz z układem napędowym.

wynosi  $\mathbf{v} = \dot{\mathbf{A}}_0^i(\mathbf{q}^i) \mathbf{r}_i$ . W konsekwencji, energia kinetyczna elementu  $dm$   $i$ -tego ramienia

$$dK_{L_i} = \frac{1}{2} dm \mathbf{v} \cdot \mathbf{v} = \frac{1}{2} dm \operatorname{tr}(\mathbf{v}\mathbf{v}^T) = \frac{1}{2} dm \operatorname{tr}\left(\dot{\mathbf{A}}_0^i(\mathbf{q}^i) \mathbf{r}_i \mathbf{r}_i^T \dot{\mathbf{A}}_0^{iT}(\mathbf{q}^i)\right).$$

Całkując energie kinetyczne wszystkich elementów ramienia nr  $i$ , wyliczamy energię kinetyczną ramienia

$$\begin{aligned} K_{L_i}(\mathbf{q}^i, \dot{\mathbf{q}}^i) &= \frac{1}{2} \operatorname{tr}\left(\dot{\mathbf{A}}_0^i(\mathbf{q}^i) \mathbf{J}_{L_i} \dot{\mathbf{A}}_0^{iT}(\mathbf{q}^i)\right) = \\ &= \frac{1}{2} \sum_{j,k=1}^n \operatorname{tr}\left(\frac{\partial \mathbf{A}_0^i}{\partial q_j}(\mathbf{q}^i) \mathbf{J}_{L_i} \left(\frac{\partial \mathbf{A}_0^i}{\partial q_k}(\mathbf{q}^i)\right)^T\right) \dot{q}_j \dot{q}_k. \end{aligned} \quad (5.8)$$

W powyższym wzorze  $\mathbf{J}_{L_i} = \int_{\text{ramię } i} \mathbf{r}_i \mathbf{r}_i^T dm$  oznacza macierz inercji  $i$ -tego ramienia. W celu obliczenia energii kinetycznej układu napędowego ramienia  $i$ -tego będziemy zakładać, że ramię  $i$ -te jest napędzane przez silnik elektryczny zamocowany na ramieniu  $(i-1)$ -szym, w przegubie  $i$ -tym, w taki sposób, że stojan silnika stanowi część ramienia  $(i-1)$ -szego, natomiast wirnik silnika obraca się wokół osi  $Z_{i-1}$  układu  $X_{i-1}Y_{i-1}Z_{i-1}$ . Z wirnikiem  $i$ -tego silnika zwiążemy układ współrzędnych  $X'_{i-1}Y'_{i-1}Z'_{i-1}$ , którego początek pokrywa się z początkiem układu  $X_{i-1}Y_{i-1}Z_{i-1}$ , obracający się

wraz z wirnikiem wokół osi  $Z_{i-1}$ . Niech  $\vartheta_i$  oznacza położenie wirnika. Wówczas, przekształcenie układów  $X_{i-1}Y_{i-1}Z_{i-1} \mapsto X'_{i-1}Y'_{i-1}Z'_{i-1}$  jest opisane macierzą  $\mathbf{Rot}(Z, \vartheta_i)$ . Transformacja układu podstawowego  $X_0Y_0Z_0$  w układ wirnika  $X'_{i-1}Y'_{i-1}Z'_{i-1}$  ma postać

$$\mathbf{A}_0^{i-1}(\mathbf{q}^{i-1}, \vartheta_i) = \mathbf{A}_0^{i-1}(\mathbf{q}^{i-1}) \mathbf{Rot}(Z, \vartheta_i).$$

W celu obliczenia energii kinetycznej wirnika wybierzemy element masy wirnika  $dm$  o współrzędnych jednorodnych  $\mathbf{s}_i = (x_i, y_i, z_i, 1)^T$  względem układu  $X'_{i-1}Y'_{i-1}Z'_{i-1}$ . Współrzędne tego elementu w układzie podstawowym są określone wzorem  $\mathbf{A}_0^{i-1}(\mathbf{q}^{i-1}) \mathbf{Rot}(Z, \vartheta_i) \mathbf{s}_i$ , a prędkość wynosi

$$\mathbf{v} = \dot{\mathbf{A}}_0^{i-1}(\mathbf{q}^{i-1}) \mathbf{Rot}(Z, \vartheta_i) \mathbf{s}_i + \mathbf{A}_0^{i-1}(\mathbf{q}^{i-1}) \dot{\mathbf{Rot}}(Z, \vartheta_i) \mathbf{s}_i.$$

Energia kinetyczna elementu  $dm$  może być przedstawiona w następującej formie

$$\begin{aligned} dK_{M_i} &= \frac{1}{2} dm \mathbf{v} \cdot \mathbf{v} = \frac{1}{2} dm \operatorname{tr}(\mathbf{v}\mathbf{v}^T) = \\ &= \frac{1}{2} dm \operatorname{tr} \left( \dot{\mathbf{A}}_0^{i-1}(\mathbf{q}^{i-1}) \mathbf{Rot}(Z, \vartheta_i) \mathbf{s}_i \mathbf{s}_i^T \mathbf{Rot}^T(Z, \vartheta_i) \left( \dot{\mathbf{A}}_0^{i-1} \right)^T(\mathbf{q}^{i-1}) \right) + \\ &+ \frac{1}{2} dm \operatorname{tr} \left( \dot{\mathbf{A}}_0^{i-1}(\mathbf{q}^{i-1}) \mathbf{Rot}(Z, \vartheta_i) \mathbf{s}_i \mathbf{s}_i^T \dot{\mathbf{Rot}}^T(Z, \vartheta_i) \left( \mathbf{A}_0^{i-1} \right)^T(\mathbf{q}^{i-1}) \right) + \\ &+ \frac{1}{2} dm \operatorname{tr} \left( \mathbf{A}_0^{i-1}(\mathbf{q}^{i-1}) \dot{\mathbf{Rot}}(Z, \vartheta_i) \mathbf{s}_i \mathbf{s}_i^T \mathbf{Rot}^T(Z, \vartheta_i) \left( \dot{\mathbf{A}}_0^{i-1} \right)^T(\mathbf{q}^{i-1}) \right) + \\ &+ \frac{1}{2} dm \operatorname{tr} \left( \mathbf{A}_0^{i-1}(\mathbf{q}^{i-1}) \dot{\mathbf{Rot}}(Z, \vartheta_i) \mathbf{s}_i \mathbf{s}_i^T \dot{\mathbf{Rot}}^T(Z, \vartheta_i) \left( \mathbf{A}_0^{i-1} \right)^T(\mathbf{q}^{i-1}) \right). \end{aligned}$$

Energię kinetyczną wirnika silnika nr  $i$  obliczymy całkując elementarne energie  $dK_{M_i}$  w obrębie wirnika,

$$\begin{aligned} K_{M_i}(\mathbf{q}^{i-1}, \vartheta_i, \dot{\mathbf{q}}^{i-1}, \dot{\vartheta}_i) &= \int_{\text{wirnik } i} dK_{M_i} = \\ &= \frac{1}{2} \operatorname{tr} \left( \dot{\mathbf{A}}_0^{i-1}(\mathbf{q}^{i-1}) \mathbf{Rot}(Z, \vartheta_i) \mathbf{J}_{M_i} \mathbf{Rot}^T(Z, \vartheta_i) \left( \dot{\mathbf{A}}_0^{i-1} \right)^T(\mathbf{q}^{i-1}) \right) + \\ &+ \operatorname{tr} \left( \dot{\mathbf{A}}_0^{i-1}(\mathbf{q}^{i-1}) \mathbf{Rot}(Z, \vartheta_i) \mathbf{J}_{M_i} \mathbf{Rot}^T(Z, \vartheta_i) \begin{bmatrix} -[\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \left( \mathbf{A}_0^{i-1} \right)^T(\mathbf{q}^{i-1}) \right) \dot{\vartheta}_i + \\ &+ \frac{1}{2} \operatorname{tr} \left( \mathbf{A}_0^{i-1}(\mathbf{q}^{i-1}) \begin{bmatrix} [\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{Rot}(Z, \vartheta_i) \mathbf{J}_{M_i} \right. \\ &\quad \left. \mathbf{Rot}^T(Z, \vartheta_i) \begin{bmatrix} -[\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \left( \mathbf{A}_0^{i-1} \right)^T(\mathbf{q}^{i-1}) \right) \dot{\vartheta}_i^2. \quad (5.9) \end{aligned}$$

W powyższej formule  $\mathbf{J}_{M_i} = \int_{\text{wirnik } i} \mathbf{s}_i \mathbf{s}_i^T dm$  oznacza macierz inercji wirnika nr  $i$ . Korzystając z własności operacji śladu macierzy i biorąc pod uwagę postać macierzy

$$\mathbf{A}_0^{i-1}(\mathbf{q}^{i-1}) = \begin{bmatrix} \mathbf{R}_0^{i-1}(\mathbf{q}^{i-1}) & \mathbf{T}_0^{i-1}(\mathbf{q}^{i-1}) \\ \mathbf{0} & 1 \end{bmatrix},$$

przekształcamy ostatni składnik wyrażenia (5.9) w następujący sposób

$$\begin{aligned} \text{tr} \left( \mathbf{A}_0^{i-1}(\mathbf{q}^{i-1}) \begin{bmatrix} [\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} \mathbf{Rot}(Z, \vartheta_i) \mathbf{J}_{M_i} \mathbf{Rot}^T(Z, \vartheta_i) \begin{bmatrix} -[\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} (\mathbf{A}_0^{i-1})^T(\mathbf{q}^{i-1}) \right) = \\ = \text{tr} \left( \mathbf{J}_{M_i} \mathbf{Rot}^T(Z, \vartheta_i) \begin{bmatrix} -[\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} (\mathbf{A}_0^{i-1})^T(\mathbf{q}^{i-1}) \mathbf{A}_0^{i-1}(\mathbf{q}^{i-1}) \begin{bmatrix} [\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} \mathbf{Rot}(Z, \vartheta_i) \right) = \\ = \text{tr} \left( \mathbf{J}_{M_i} \begin{bmatrix} -[\mathbf{e}_3]^2 & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} \right) = \int_{\text{wirnik } i} (x_i^2 + y_i^2) dm = I_i, \end{aligned}$$

gdzie  $I_i$  oznacza moment bezwładności wirnika silnika względem osi  $Z_{i-1}$  układu  $X_{i-1}Y_{i-1}Z_{i-1}$ . Pokazaliśmy, że trzeci składnik energii kinetycznej wirnika ma postać

$$\frac{1}{2} I_i \dot{\vartheta}_i^2.$$

Wykonując różniczkowanie względem czasu w wyrażeniu (5.9), otrzymujemy ostatecznie

$$\begin{aligned} K_{M_i}(\mathbf{q}^{i-1}, \vartheta_i, \dot{\mathbf{q}}^{i-1}, \dot{\vartheta}_i) = \\ = \frac{1}{2} \sum_{j,k=1}^n \text{tr} \left( \frac{\partial \mathbf{A}_0^{i-1}}{\partial q_j}(\mathbf{q}^{i-1}) \mathbf{Rot}(Z, \vartheta_i) \mathbf{J}_{M_i} \mathbf{Rot}^T(Z, \vartheta_i) \left( \frac{\partial \mathbf{A}_0^{i-1}}{\partial q_k}(\mathbf{q}^{i-1}) \right)^T \right) \dot{q}_j \dot{q}_k + \\ + \sum_{j=1}^n \text{tr} \left( \frac{\partial \mathbf{A}_0^{i-1}}{\partial q_j}(\mathbf{q}^{i-1}) \mathbf{Rot}(Z, \vartheta_i) \mathbf{J}_{M_i} \mathbf{Rot}^T(Z, \vartheta_i) \begin{bmatrix} -[\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} (\mathbf{A}_0^{i-1})^T(\mathbf{q}^{i-1}) \right) \dot{q}_j \dot{\vartheta}_i + \\ + \frac{1}{2} I_i \dot{\vartheta}_i^2. \quad (5.10) \end{aligned}$$

W manipulatorze sztywnym transmisja napędu do przegubu jest opisana przez przełożenie przekładni,  $\kappa_i q_i = \vartheta_i$ . Biorąc to pod uwagę oraz wprowadzając oznaczenie

$$\mathbf{J}_{M_i}^* = \mathbf{Rot}(Z, \vartheta_i) \mathbf{J}_{M_i} \mathbf{Rot}^T(Z, \vartheta_i),$$

zapisujemy energię kinetyczną (5.10) w postaci

$$\begin{aligned}
 K_{M_i}(\mathbf{q}^i, \dot{\mathbf{q}}^i) = & \\
 = \frac{1}{2} \sum_{j,k=1}^n \operatorname{tr} \left( \frac{\partial \mathbf{A}_0^{i-1}}{\partial \mathbf{q}_j}(\mathbf{q}^{i-1}) \mathbf{J}_{M_i}^* \left( \frac{\partial \mathbf{A}_0^{i-1}}{\partial \mathbf{q}_k}(\mathbf{q}^{i-1}) \right)^T \right) \dot{q}_j \dot{q}_k + & \\
 + \sum_{j=1}^n \operatorname{tr} \left( \frac{\partial \mathbf{A}_0^{i-1}}{\partial \mathbf{q}_j}(\mathbf{q}^{i-1}) \mathbf{J}_{M_i}^* \begin{bmatrix} -[\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} (\mathbf{A}_0^{i-1})^T(\mathbf{q}^{i-1}) \right) \kappa_i \dot{q}_j \dot{q}_i + & \\
 & + \frac{1}{2} \kappa_i^2 I_i \dot{q}_i^2. \quad (5.11)
 \end{aligned}$$

Sumaryczna energia kinetyczna wszystkich ramion i układów napędowych manipulatora sztywnego jest opisana formułą

$$K(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{Q}(\mathbf{q}) \dot{\mathbf{q}}, \quad (5.12)$$

z symetryczną, dodatnio określoną macierzą  $\mathbf{Q}(\mathbf{q})$ , której elementy mają postać wynikającą z wyrażeń (5.8), (5.11), mianowicie

$$\begin{aligned}
 Q_{jk}(\mathbf{q}) = \sum_{i=1}^{n-1} \operatorname{tr} \left( \frac{\partial \mathbf{A}_0^i}{\partial \mathbf{q}_j}(\mathbf{q}^i) (\mathbf{J}_{L_i} + \mathbf{J}_{M_{i+1}}^*) \left( \frac{\partial \mathbf{A}_0^i}{\partial \mathbf{q}_k}(\mathbf{q}^i) \right)^T \right) + & \\
 + \operatorname{tr} \left( \frac{\partial \mathbf{A}_0^n}{\partial \mathbf{q}_j}(\mathbf{q}) \mathbf{J}_{L_n} \left( \frac{\partial \mathbf{A}_0^n}{\partial \mathbf{q}_k}(\mathbf{q}) \right)^T \right) + & \\
 + \operatorname{tr} \left( \frac{\partial \mathbf{A}_0^{k-1}}{\partial \mathbf{q}_j}(\mathbf{q}^{k-1}) \mathbf{J}_{M_k}^* \begin{bmatrix} -[\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} (\mathbf{A}_0^{k-1})^T(\mathbf{q}^{k-1}) \right) \kappa_k + \kappa_j^2 I_j \delta_{jk}. \quad (5.13)
 \end{aligned}$$

We wzorze (5.13)  $\delta_{ij}$  oznacza funkcję zwaną *deltą Kroneckera*, określoną formułą

$$\delta_{ij} = \begin{cases} 1, & \text{jeżeli } i = j, \\ 0, & \text{jeżeli } i \neq j. \end{cases}$$

Dla  $i = 0$  macierz  $\mathbf{A}_0^i$  definiujemy jako macierz jednostkową  $\mathbb{I}_4$ .

Energia potencjalna manipulatora pochodzi od oddziaływania pola grawitacyjnego Ziemi. W celu obliczenia energii potencjalnej ramienia nr  $i$  (wraz z układem napędowym), można je potraktować jako masę punktową  $m_i$  skupioną w środku masy ramienia. Zakładając, że  $\mathbf{R}_i$  oznacza

współrzędne jednorodnie środka masy w układzie  $X_i Y_i Z_i$ , obliczamy energię potencjalną  $i$ -tego ramienia jako

$$V_i = -m_i \mathbf{g}^T \mathbf{A}_0^i(\mathbf{q}^i) \mathbf{R}_i, \quad (5.14)$$

gdzie  $\mathbf{g} = (g_1, g_2, g_3, 0)^T$  jest wektorem przyspieszenia ziemskiego w układzie podstawowym. Całkowita energia potencjalna manipulatora

$$V(\mathbf{q}) = - \sum_{i=1}^n m_i \mathbf{g}^T \mathbf{A}_0^i(\mathbf{q}^i) \mathbf{R}_i. \quad (5.15)$$

Forma energii kinetycznej definiuje dwa pierwsze elementy modelu dynamiki (5.7) (macierz inercji  $\mathbf{Q}(\mathbf{q})$  i macierz sił odśrodkowych i sił Coriolisa  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ ), natomiast gradient energii potencjalnej  $\frac{\partial V}{\partial \mathbf{q}}(\mathbf{q})$  wyznacza wektor sił grawitacji  $\mathbf{D}(\mathbf{q})$ . Określenie składnika niepotencjalnego  $\zeta(\mathbf{q}, \dot{\mathbf{q}}, t)$  wymaga specjalnych badań. Jeżeli składnik ten nie występuje, równania dynamiki manipulatora sztywnego uzyskują postać

$$\mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) = \mathbf{u}. \quad (5.16)$$

Można pokazać, że poszczególne elementy modelu dynamiki (5.16), oprócz własności ogólnych (5.6), mają następujące własności

$$\|\mathbf{Q}(\mathbf{q})\| \leq Q_M, \quad (5.17)$$

$$\mathbf{C}(\mathbf{q}, \mathbf{x})\mathbf{y} = \mathbf{C}(\mathbf{q}, \mathbf{y})\mathbf{x}, \quad (5.18)$$

$$\|\mathbf{C}(\mathbf{q}, \mathbf{x})\| \leq C_M \|\mathbf{x}\|, \quad (5.19)$$

$$\|\mathbf{D}(\mathbf{q})\| \leq D_M, \quad (5.20)$$

przy czym  $Q_M, C_M, D_M > 0$  są pewnymi stałymi, a własność (5.20) zachodzi dla manipulatorów o przegubach obrotowych lub ograniczonych przegubach przesuwanych. W powyższych zależnościach norma wektora powinna być rozumiana jako norma euklidesowa, natomiast jako normę macierzy bierzemy normę indukowaną przez normę euklidesową

$$\|\mathbf{A}\| = \sqrt{\bar{\lambda}_{\mathbf{A}^T \mathbf{A}}},$$

gdzie  $\bar{\lambda}_{\mathbf{M}}$  oznacza największą wartość własną symetrycznej macierzy  $\mathbf{M}^*$ .

\*Zobacz dodatek A.1.

**Przykład 5.2.1 (Manipulator EDDA)**

Rozpatrzmy manipulator *EDDA*, którego model kinematyki został opisany w przykładzie 2.3.8. Dynamika tego manipulatora może być wyrażona równaniem (5.16), przy czym macierz inercji, macierz sił Coriolisa i wektor grawitacji mają postać

$$\mathbf{Q}(\mathbf{q}) = \begin{bmatrix} \theta_1 + 0.6\theta_4 \cos q_2 & \theta_3 + 0.3\theta_4 \cos q_2 \\ \theta_3 + 0.3\theta_4 \cos q_2 & \theta_3 \end{bmatrix}, \quad (5.21)$$

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} -\dot{q}_1 & -(\dot{q}_1 + \dot{q}_2) \\ \dot{q}_1 & 0 \end{bmatrix} 0.3\theta_4 \sin q_2, \quad (5.22)$$

$$\mathbf{D}(\mathbf{q}) = g \begin{pmatrix} \theta_2 \cos q_1 + \theta_4 \cos(q_1 + q_2) \\ \theta_4 \cos(q_1 + q_2) \end{pmatrix}. \quad (5.23)$$

Parametry modelu dynamiki przyjmują następujące wartości nominalne

$$\begin{aligned} \theta_1 &= I_1 + I_2 + m_2 l_1^2 = 3.1 [\text{kg} \cdot \text{m}^2], \\ \theta_2 &= m_1 p_1 + m_2 l_1 = 9.5 [\text{kg} \cdot \text{m}], \\ \theta_3 &= I_2 = 0.24 [\text{kg} \cdot \text{m}^2], \\ \theta_4 &= m_2 p_2 = 0.77 [\text{kg} \cdot \text{m}], \end{aligned} \quad (5.24)$$

przy oznaczeniach

- $m_i$  — masa  $i$ -tego ogniwa,
- $p_i$  — położenie środka masy  $i$ -tego ogniwa w  $i$ -tym układzie współrzędnych,
- $l_i$  — długość  $i$ -tego ogniwa,
- $I_i$  — moment bezwładności  $i$ -tego ogniwa (względem osi  $Z$  układu  $X_i Y_i Z_i$ ),
- $g$  — wartość przyspieszenia ziemskiego. ■

### 5.3 Sterowanie w przestrzeni zadaniowej manipulatora

W niniejszym rozdziale rozważymy zadanie polegające na śledzeniu trajektorii przez manipulator sztywny przy założeniu, że zadana jest trajektoria w przestrzeni zadaniowej manipulatora  $\mathbf{y}_d(t) \in \mathcal{Z} \subset \mathbb{R}^m$  co oznacza, że efektor manipulatora ma się poruszać wzdłuż zadanej krzywej w przestrzeni, zachowując określoną orientację. Przyjmujemy, że model manipulatora obejmuje zarówno równania dynamiki, jak i równania kinematyki

(odwzorowanie stan-wyjście), i ma postać

$$\begin{cases} \mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) = \mathbf{u} \\ \mathbf{y} = \mathbf{k}(\mathbf{q}). \end{cases} \quad (5.25)$$

### 5.3.1 Linearyzacja i odsprzężanie wejściowo-wyjściowe

Wprowadźmy w układzie (5.25) nowe współrzędne

$$\begin{cases} \mathbf{x} = \mathbf{q} \\ \boldsymbol{\xi} = \dot{\mathbf{q}}, \end{cases}$$

i wyrażmy równania układu w tych współrzędnych w postaci afinicznego układu sterowania z funkcją wyjścia

$$\begin{cases} \dot{\mathbf{x}} = \boldsymbol{\xi} \\ \dot{\boldsymbol{\xi}} = \mathbf{F}(\mathbf{x}, \boldsymbol{\xi}) + \mathbf{G}(\mathbf{x})\mathbf{u} \\ \mathbf{y} = \mathbf{k}(\mathbf{x}), \end{cases} \quad (5.26)$$

gdzie  $\mathbf{F}(\mathbf{x}, \boldsymbol{\xi}) = -\mathbf{Q}^{-1}(\mathbf{x})(\mathbf{C}(\mathbf{x}, \boldsymbol{\xi})\boldsymbol{\xi} + \mathbf{D}(\mathbf{x}))$ ,  $\mathbf{G}(\mathbf{x}) = \mathbf{Q}^{-1}(\mathbf{x})$ . Przyjmijmy dodatkowo, że manipulator jest nieredundantny, a zatem liczba stopni swobody manipulatora (i liczba wejść sterujących) jest równa wymiarowi wektora współrzędnych położenia i orientacji efektora, czyli liczbie wyjść (tzn.  $n = m$ ). Idea algorytmu sterowania przedstawionego poniżej opiera się na dwóch elementach. Pierwszym jest *odsprzężenie* i *linearyzacja* modelu manipulatora, czyli uzyskanie takiego nieliniowego *sprzężenia zwrotnego*, które po zastosowaniu do modelu manipulatora spowoduje, że *i*-te sterowanie będzie oddziaływało wyłącznie na *i*-tą składową wektora wyjściowego, natomiast odwzorowanie wejście-wyjście stanie się liniowe. Drugi element algorytmu sterowania jest standardowy i sprowadza się do zastosowania w odsprzężonym modelu manipulatora liniowego regulatora PD z korekcją.

Konstrukcję odsprzęgającego i linearyzującego sprzężenia zwrotnego dla modelu (5.26) rozpoczniemy od wyliczenia kolejnych pochodnych czasowych wyjść  $y_i$  wzdłuż trajektorii układu. Dla wyjścia  $y_i$  otrzymujemy

$$\begin{aligned} y_i(t) &= k_i(\mathbf{x}(t)), \\ \dot{y}_i(t) &= \frac{\partial k_i(\mathbf{x})}{\partial \mathbf{x}} \dot{\mathbf{x}} = \frac{\partial k_i(\mathbf{x})}{\partial \mathbf{x}} \boldsymbol{\xi}. \end{aligned}$$

Zauważmy, że  $\ddot{y}_i$  nie zależy bezpośrednio od sterowania. Policzmy pochodną rzędu drugiego

$$\begin{aligned}\ddot{y}_i(t) &= \frac{\partial}{\partial \mathbf{x}} \left( \frac{\partial k_i(\mathbf{x})}{\partial \mathbf{x}} \boldsymbol{\xi} \right) \dot{\boldsymbol{\xi}} + \frac{\partial k_i(\mathbf{x})}{\partial \mathbf{x}} \ddot{\boldsymbol{\xi}} = \\ &= \boldsymbol{\xi}^\top \frac{\partial^2 k_i(\mathbf{x})}{\partial \mathbf{x}^2} \boldsymbol{\xi} + \frac{\partial k_i(\mathbf{x})}{\partial \mathbf{x}} (\mathbf{F}(\mathbf{x}, \boldsymbol{\xi}) + \mathbf{G}(\mathbf{x})\mathbf{u}) = \\ &= \boldsymbol{\xi}^\top \frac{\partial^2 k_i(\mathbf{x})}{\partial \mathbf{x}^2} \boldsymbol{\xi} + \frac{\partial k_i(\mathbf{x})}{\partial \mathbf{x}} \mathbf{F}(\mathbf{x}, \boldsymbol{\xi}) + \frac{\partial k_i(\mathbf{x})}{\partial \mathbf{x}} \mathbf{G}(\mathbf{x})\mathbf{u}.\end{aligned}$$

Zapisując analogiczne równania dla  $i = 1, \dots, n$  otrzymujemy następujący układ równań

$$\ddot{\mathbf{y}} = \mathbf{P}(\mathbf{x}, \boldsymbol{\xi}) + \mathbf{H}(\mathbf{x})\mathbf{u}, \quad (5.27)$$

w którym

$$\begin{aligned}P_i(\mathbf{x}, \boldsymbol{\xi}) &= \boldsymbol{\xi}^\top \frac{\partial^2 k_i(\mathbf{x})}{\partial \mathbf{x}^2} \boldsymbol{\xi} + \frac{\partial k_i(\mathbf{x})}{\partial \mathbf{x}} \mathbf{F}(\mathbf{x}, \boldsymbol{\xi}), \\ \mathbf{H}(\mathbf{x}) &= \frac{\partial \mathbf{k}(\mathbf{x})}{\partial \mathbf{x}} \mathbf{G}(\mathbf{x}).\end{aligned}$$

W dalszych rozważaniach przyjmiemy założenie, że macierz  $\mathbf{H}(\mathbf{x})$  jest nieosobliwa dla każdego  $\mathbf{x}$ . Jest ono spełnione wtedy, gdy jacobian analityczny  $\mathbf{J}^a(\mathbf{x}) = \frac{\partial \mathbf{k}}{\partial \mathbf{x}}(\mathbf{x})$  jest pełnego rzędu, a więc gdy manipulator nie przyjmuje konfiguracji osobliwych (przy realizacji zadanej trajektorii  $\mathbf{y}_d(t)$ ). Ponieważ macierz  $\mathbf{G}(\mathbf{x})$  jest zawsze odwracalna, przyjęte założenie pozwala zastosować do układu (5.27) sprzężenie zwrotne

$$\mathbf{u} = -\mathbf{H}^{-1}(\mathbf{x})\mathbf{P}(\mathbf{x}, \boldsymbol{\xi}) + \mathbf{H}^{-1}(\mathbf{x})\mathbf{v} \quad (5.28)$$

i przekształcić go do postaci

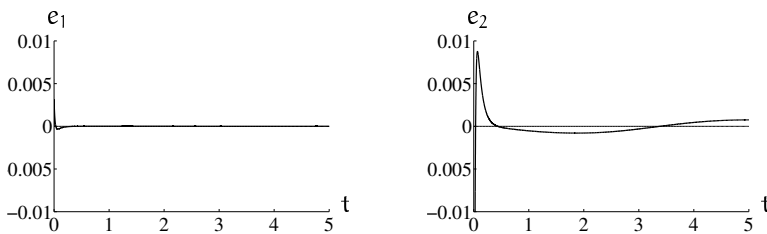
$$\ddot{y}_i = v_i, \quad i = 1, \dots, n, \quad (5.29)$$

co oznacza pełne odsprężenie wejść i wyjść oraz liniowość odwzorowania wejście-wyjście.

Po uzyskaniu liniowego i odsprężonego modelu manipulatora będziemy szukać sterowania  $\mathbf{v}(t)$  generującego trajektorię  $\mathbf{y}(t)$  efektora w taki sposób, aby błąd śledzenia

$$\mathbf{e}(t) = \mathbf{y}(t) - \mathbf{y}_d(t) \quad (5.30)$$





**Rysunek 5.2** Przebieg błędów śledzenia przy zastosowaniu algorytmu odsprężania wejściowo-wyjściowego ze wzmocnieniami  $\mathbf{K}_p = \text{diag}\{10000\}$  i  $\mathbf{K}_d = \text{diag}\{1000\}$ .

dążył asymptotycznie do zera. W celu osiągnięcia żądanego zachowania manipulatora, do każdego wejścia  $v_i$  dołączymy lokalny regulator typu PD z korekcją co oznacza, że sygnał podawany na wejście układu (5.29) będzie miał postać

$$\mathbf{v} = \ddot{\mathbf{y}}_d - \mathbf{K}_d(\dot{\mathbf{y}} - \dot{\mathbf{y}}_d) - \mathbf{K}_p(\mathbf{y} - \mathbf{y}_d). \quad (5.31)$$

Macierze  $\mathbf{K}_d = \text{diag}\{k_{di}\} > 0$  i  $\mathbf{K}_p = \text{diag}\{k_{pi}\} > 0$ ,  $i = 1, \dots, n$ , są diagonalnymi macierzami wzmocnień lokalnych regulatorów PD. Taki wybór macierzy wzmocnień gwarantuje, że błąd regulacji określony równaniem różniczkowym

$$\ddot{\mathbf{e}} + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e} = \mathbf{0}$$

zanika do zera eksponencjalnie, gdy  $t$  dąży do  $+\infty$ .

### Przykład 5.3.1 (Manipulator *EDDA*)

Działanie algorytmu odsprężania wejściowo-wyjściowego zilustrujemy na przykładzie manipulatora *EDDA*. Niech zadanie sterowania polega na śledzeniu trajektorii cyklicznej wyrażonej w przestrzeni zadaniowej manipulatora (w aspekcie współrzędnych  $x_0, y_0$ ) równaniem

$$\mathbf{y}_d(t) = \begin{pmatrix} y_{d1}(t) \\ y_{d2}(t) \end{pmatrix} = \begin{pmatrix} 0.4 + 0.1 \cos t \\ -0.6 + 0.1 \sin t \end{pmatrix}.$$

Rysunek 5.2 przedstawia wykresy błędu (5.30) śledzenia współrzędnych położenia efektora\*. Symbol  $e_1$  oznacza błąd śledzenia współrzędnej  $y_1$ , natomiast symbol  $e_2$  — błąd śledzenia współrzędnej  $y_2$ . ■

Z przeprowadzonych symulacji wynika, że przystępując do implementacji algorytmu odsprężania, należy zwrócić szczególną uwagę na zapewnienie

\*Symulacje wykonano w środowisku MATLAB® + SIMULINK®.

odwracalności jacobianu analitycznego manipulatora. Praktycznie oznacza to nie tylko, że przy realizacji trajektorii zadanej manipulator nie przyjmuje konfiguracji osobliwych, lecz także nie zbliża się do nich. Algorytm odsprężania jest trudny do implementacji ze względu na dużą wrażliwość na warunki początkowe, wybór metody całkowania, a także na wartości wzmocnień  $K_p$ ,  $K_d$ .

### 5.3.2 Transformacja do zadania sterowania w przestrzeni przegubowej

Większość algorytmów sterowania manipulatorów sztywnych dotyczy sterowania w przestrzeni przegubowej. Aby zastosować takie algorytmy do sterowania w przestrzeni zadaniowej manipulatora, niezbędne jest przeniesienie trajektorii zadanej, określonej w naturalny sposób w przestrzeni zadaniowej, do przestrzeni przegubowej. Polega ono na rozwiązaniu odwrotnego zadania kinematyki i wyliczeniu  $n$ -wymiarowego wektora trajektorii zadanych zdefiniowanych dla każdego przegubu manipulatora. Potrzebne algorytmy kinematyki odwrotnej manipulatora zostały omówione w rozdziale 3.

## 5.4 Komentarze i uwagi bibliograficzne

Klasyczne układy sterowania manipulatorów można znaleźć w większości podręczników [Pau81, Cra93, SV97]. Doniesienie o wdrożeniu algorytmów sterowania opartych na modelu dynamiki manipulatora przez firmę KUKA pochodzi od Hirzingera [H<sup>+</sup>98]. Formalizm lagranżowski i hamiltonowski mechaniki analitycznej stanowi przedmiot klasycznego wykładu mechaniki [RK95, AM78, Arn78]. Wyprowadzenie równań dynamiki manipulatora przedstawione w podrozdziale 5.2 jest również klasyczne [Pau81, AS86, SV97]. Identyfikacja parametrów modeli kinematyki (zwana kalibracją kinematyki) i dynamiki manipulatorów należy do podstawowych zadań robotyki [SS96, Koz98, MRD91]. Bardziej wnikliwe potraktowanie dynamiki układów napędowych okaże się przydatne w rozdziale poświęconym modelowi dynamiki manipulatora o elastycznych przegubach. Idea sterowania w przestrzeni zewnętrznej z odsprężeniem modelu pochodzi od Freunda [Fre82]. Metoda ta jest znana w geometrycznej teorii sterowania pod nazwą metody odsprężania wejściowo-wyjściowego [Isi89, NS90]. Określenia interakcji dynamicznych i odsprężenia manipulatora IRb-6 dokonano w pracy [GW86].

## Literatura

- [AM78] R. Abraham i J. E. Marsden, *Foundations of Mechanics*. The Benjamin, Reading, 1978.
- [Arn78] V. I. Arnold, *Mathematical Methods of Classical Mechanics*. Springer-Verlag, Berlin, 1978.
- [AS86] H. Asada i J. J. E. Slotine, *Robot Analysis and Control*. J. Wiley and Sons, New York, 1986.
- [Cra93] J. J. Craig, *Wprowadzenie do robotyki*. WNT, Warszawa, 1993.
- [Fre82] E. Freund, Fast nonlinear control with arbitrary pole-placement for industrial robots and manipulators. *Int. J. Robotics Research*, 1(1):65–78, 1982.
- [GW86] A. Gosiewski i W. Wieczorek, Dynamic interactions in IRb robot. *Control & Cybernetics*, 15(3–4):462–472, 1986.
- [H<sup>+</sup>98] G. Hirzinger et al., Towards a new robot generation. W: *Proc. MMAR Symposium*, vol. 3, strony 747–762, Międzyzdroje, 1998.
- [Isi89] A. Isidori, *Nonlinear Control Systems*. Springer-Verlag, Berlin, 1989.
- [Koz98] K. Kozłowski, *Modelling and Identification in Robotics*. Springer-Verlag, Berlin, 1998.
- [MRD91] B. W. Mooring, Z. S. Roth i M. R. Driels, *Fundamentals of Manipulator Calibration*. J. Wiley and Sons, New York, 1991.
- [NS90] H. Nijmeijer i A. J. van der Schaft, *Nonlinear Dynamic Control Systems*. Springer-Verlag, New York, 1990.
- [Pau81] R. P. Paul, *Robot Manipulators: Mathematics, Programming and Control*. MIT Press, Cambridge, 1981.
- [RK95] W. Rubinowicz i W. Królikowski, *Mechanika teoretyczna*. PWN, Warszawa, 1995.
- [SS96] L. Sciavicco i B. Siciliano, *Modeling and Control of Robot Manipulators*. The McGraw-Hill, New York, 1996.
- [SV97] M. Spong i M. Vidyasagar, *Dynamika i sterowanie robotów*. WNT, Warszawa, 1997.



## Rozdział 6

# Algorytmy sterowania manipulatorów sztywnych w przestrzeni przegubowej

W niniejszym rozdziale przedstawimy algorytmy sterowania manipulatorów sztywnych wykorzystujące różny stopień znajomości matematycznego opisu zachowania manipulatora. W przypadku, gdy znany jest model dynamiki manipulatora, przy syntezie algorytmów sterowania stosuje się dwa podejścia. Pierwsze podejście polega na wykorzystaniu odwracalności macierzy bezwładności manipulatora i prowadzi do algorytmów typu *obliczanego momentu*. Z punktu widzenia geometrycznej teorii sterowania, metoda obliczanego momentu stanowi przykład linearyzacji dynamiki układu przez statyczne sprzężenie zwrotne. W zasadzie, procedura linearyzacji modelu manipulatora przy pomocy sprzężenia zwrotnego pozwala usunąć z dynamiki manipulatora wszystkie składniki nieliniowe, ale z praktycznego punktu widzenia posiada ona istotne ograniczenia polegające na wymaganiu pełnej znajomości modelu dynamiki oraz konieczności obliczania modelu w trybie *on-line*.

Drugie podejście prowadzi do algorytmów sterowania typu *dysypatywnego*. W przeciwieństwie do algorytmów typu obliczanego momentu, algorytmy typu dysypatywnego nie wymagają sprzężenia linearyzującego. Działanie tych algorytmów polega na rozpraszaniu energii układu w taki sposób, aby dla układu sterowania z zamkniętą pętlą sprzężenia zwrotnego błąd śledzenia położenia i prędkości przegubów manipulatora zmierzał asymptotycznie do zera.

Załóżmy, że model dynamiki manipulatora ma postać

$$\mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) + \boldsymbol{\zeta}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{u}. \quad (6.1)$$

Model (6.1) różni się od modelu (5.16) obecnością wektora  $\boldsymbol{\zeta}(\mathbf{q}, \dot{\mathbf{q}}, t)$ , który oznacza różnego rodzaju siły niepotencjalne, włączając efekty wynikające z niedokładności modelowania. Algorytmy typu obliczanego momentu i typu dysypatywnego wymagają pełnej znajomości modelu manipulatora. Oczywiście, w praktyce model dynamiki manipulatora jest zawsze obciążony niepewnością. Nieznajomość modelu można podzielić na dwa rodzaje: nieznajomość *parametryczną* i nieznajomość *strukturalną*. Nieznajomość parametryczna polega na tym, że znane są wszystkie składniki równań modelu (znana jest ich postać funkcyjna), natomiast nie są znane pewne parametry tych równań. Model manipulatora zawiera wiele parametrów, które zależą od własności fizycznych manipulatora, takich jak długości ramion oraz masy i momenty bezwładności ramion, a także od własności przenoszonego ładunku, itp. Zauważmy, że zmienna masa i moment bezwładności przenoszonego ładunku powoduje zmiany parametrów modelu dynamiki w trakcie pracy manipulatora.

Innego rodzaju nieznajomość modelu stanowi nieznajomość strukturalna, która polega na tym, że nie jest znana zależność funkcyjna między pewnymi zmiennymi modelu. Nieznajomość strukturalna modelu jest często skutkiem niedokładności modelowania dynamiki manipulatora. W modelu (6.1) wszystkie nieznanne oddziaływania, którym podlega manipulator, mieszczą się w składniku  $\boldsymbol{\zeta}(\mathbf{q}, \dot{\mathbf{q}}, t)$ . Oddziaływania te zostaną potraktowane jako zaburzenia modelu podstawowego (5.3).

Ponieważ z reguły model dynamiki manipulatora nie jest dokładnie znany, szczególne zainteresowanie robotyków budzą algorytmy sterowania zapewniające śledzenie trajektorii pomimo nieznajomości modelu. W teorii sterowania są znane dwa rozwiązania tego problemu: algorytmy *sterowania adaptacyjnego* i algorytmy *sterowania odpornego*. Sterowanie adaptacyjne polega na zaprojektowaniu układu sterowania, który dostosowuje swoje działanie do zmian obiektu sterowania w taki sposób, aby cel sterowania został osiągnięty. Typowy adaptacyjny algorytm sterowania wykorzystuje układ estymujący nieznanne parametry modelu i generuje sygnał sterujący w oparciu o otrzymane estymaty parametrów. Z tego względu, układy adaptacyjne są najczęściej wykorzystywane w sytuacji, gdy model jest obciążony nieznajomością parametryczną. Układy adaptacyjne są skomplikowane, zwłaszcza w przypadku, gdy zawierają układ estymacji. Są

one skuteczne, lecz kosztowne i trudne do realizacji. Opracowano wiele algorytmów sterowania manipulatora, które są adaptacyjnymi wersjami algorytmów typu obliczanego momentu i algorytmów typu dysypatywnego.

Konkurencyjne rozwiązanie problemu sterowania przy niepełnej znajomości modelu stanowią algorytmy sterowania odporne\*. Algorytmy odporne wykorzystują pewne aprioryczne oszacowania parametrów modelu i w oparciu o taką informację zapewniają poprawne sterowanie dla wszystkich dopuszczalnych zaburzeń modelu. Ze względu na brak układu estymacji odporne układy sterowania są znacznie prostsze niż adaptacyjne, cechują się jednak strukturalną nadczynnością (np. nadmiernym wzmocnieniem) dla wartości parametrów innych niż ekstremalne.

W połowie lat osiemdziesiątych w dziedzinie sterowania adaptacyjnego pojawił się nowy paradygmat, który umożliwił opracowanie prostych algorytmów sterowania zapewniających osiągnięcie celu sterowania dla całej klasy obiektów na podstawie znajomości sygnałów wyjściowych obiektu. Dla układów liniowych minimalnofazowych odpowiednie układy sterowania zostały nazwane *uniwersalnymi adaptacyjnymi układami sterowania* (w skrócie UAUS). Układy uniwersalne są adaptacyjne w tym sensie, że realizują zadanie sterowania korzystając z ograniczonej informacji o obiekcie (jedynie informacje, jakie posiada układ sterowania, to informacje strukturalne na temat liniowości i minimalnofazowości obiektu sterowania), nie zawierają jednak układu estymacji parametrów. Uważa się, że sposób działania uniwersalnych układów sterowania imituje postępowanie człowieka, który ma sterować obiektem o nieznanym parametrach. Wyobraźmy sobie człowieka, który próbuje balansować tyczką ustawioną na dłoni. Na początku jego ruchy, mające na celu utrzymanie tyczki w pozycji pionowej, są przesadnie obszerne, lecz po krótkim okresie prób i błędów ich amplituda i częstotliwość maleje i tyczka jest trwale utrzymywana w pobliżu położenia równowagi. Po zmianie tyczki na inną wynik doświadczenia jest podobny. Uniwersalne adaptacyjne układy sterowania charakteryzują się prostą budową, brakiem układu estymacji i skutecznością działania.

Działanie uniwersalnego adaptacyjnego układu sterowania można zilustrować na przykładzie zadania stabilizacji obiektu liniowego. Każdy układ należący do klasy minimalnofazowych układów liniowych, jednowejściowych

---

\*Zastosowany tu termin „odporny” odpowiada, przyjętemu w środowisku automatyków tłumaczeniu angielskiego przymiotnika „robust”. Spośród alternatywnych jego przekładów, zaproponowanych przez robotyków, najbardziej przypadła nam do gustu propozycja „chędogi”, autorstwa dra hab. A. Kasińskiego.

i jednowyjściowych, o stopniu względnym równym 1, postaci

$$\begin{cases} \dot{\mathbf{q}}(t) = \mathbf{A}\mathbf{q}(t) + \mathbf{b}u(t) \\ \mathbf{y}(t) = \mathbf{c}^T\mathbf{q}(t), \end{cases} \quad (6.2)$$

gdzie  $\mathbf{A} \in \mathbb{R}^{n \times n}$ ,  $\mathbf{b} \in \mathbb{R}^n$ ,  $\mathbf{c} \in \mathbb{R}^n$ ,  $\mathbf{c}^T\mathbf{b} > 0$ , daje się ustabilizować za pomocą dynamicznego sprzężenia zwrotnego od wyjścia

$$\begin{cases} u(t) = -k(t)y(t) \\ \dot{k}(t) = y^2(t). \end{cases}$$

Można by oczekiwać, że znajomość znaku wyrażenia  $\mathbf{c}^T\mathbf{b}$  jest warunkiem koniecznym zbieżności każdego adaptacyjnego algorytmu stabilizacji. Jednak, dzięki wprowadzeniu pojęcia funkcji przełączającej, okazało się możliwe zaproponowanie adaptacyjnego układu stabilizacji, pomimo że znak wyrażenia  $\mathbf{c}^T\mathbf{b}$  jest nieznany. W rezultacie, jeżeli tylko  $\mathbf{c}^T\mathbf{b} \neq 0$ , układy postaci (6.2) można ustabilizować przy użyciu uniwersalnego adaptacyjnego układu sterowania

$$\begin{cases} u(t) = N(k(t))y(t) \\ \dot{k}(t) = y^2(t), \end{cases}$$

gdzie  $N(\cdot)$  jest tzw. *funkcją Nussbauma*, to znaczy funkcją odcinkami prawostronnie ciągłą, spełniającą warunki

$$\begin{aligned} \inf_{k>0} \frac{1}{k} \int_0^k N(s) ds &= -\infty, \\ \sup_{k>0} \frac{1}{k} \int_0^k N(s) ds &= +\infty. \end{aligned}$$

Przykładowymi funkcjami Nussbauma są funkcje  $N(k) = k^2 \cos k$  i  $N(k) = \cos(k\pi/2)e^{k^2}$ . Mechanizm sterowania jest następujący: układ adaptacji opisany równaniem  $\dot{k} = y^2$  monotonicznie zwiększa  $k(t)$ , a funkcja przełączająca  $N(k)$  przyjmuje na przemian wartości dodatnie i ujemne o rosnącej amplitudzie i czasie trwania. Jeśli poprawny znak funkcji  $N$  jest stały w czasie dostatecznie długim na to, aby sygnał wyjściowy  $y$  obiektu z zamkniętą pętlą sprzężenia zwrotnego zmniejszył się do zera, to adaptacja wzmocnienia dynamicznego  $k$  zatrzyma się na pewnej skończonej wartości.



## 6.1 Algorytmy wymagające pełnej znajomości modelu

### 6.1.1 Algorytmy typu obliczanego momentu

#### Dokładna linearyzacja

Najbardziej znany algorytm typu obliczanego momentu korzysta z linearyzacji modelu dynamiki manipulatora przy pomocy statycznego sprzężenia zwrotnego. Aby przedstawić ideę tego algorytmu, rozważmy model dynamiki manipulatora sztywnego

$$\mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) = \mathbf{u}. \quad (6.3)$$

Na wejście układu podajmy sygnał sterujący

$$\mathbf{u} = \mathbf{Q}(\mathbf{q})\mathbf{v} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}), \quad (6.4)$$

gdzie  $\mathbf{v}$  jest nowym wektorem sterowań. Po podstawieniu sterowania (6.4) do modelu dynamiki manipulatora (6.3) otrzymujemy równanie układu z zamkniętą pętlą sprzężenia zwrotnego postaci

$$\mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}} = \mathbf{Q}(\mathbf{q})\mathbf{v},$$

lub równoważne mu równanie liniowe

$$\ddot{\mathbf{q}} = \mathbf{v}, \quad (6.5)$$

uzyskane dzięki odwracalności macierzy bezwładności  $\mathbf{Q}(\mathbf{q})$ .

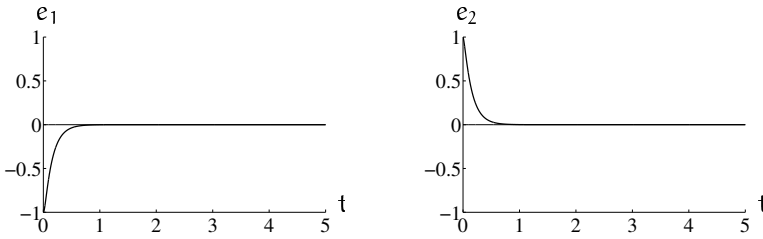
W celu zapewnienia śledzenia trajektorii  $\mathbf{q}_d(t)$  w układzie zlinearyzowanym, wystarczy podać na wejście  $\mathbf{v}$  sygnał z regulatora PD z korekcją

$$\mathbf{v} = \ddot{\mathbf{q}}_d - \mathbf{K}_d\dot{\mathbf{e}} - \mathbf{K}_p\mathbf{e}, \quad (6.6)$$

gdzie  $\mathbf{e}(t) = \mathbf{q}(t) - \mathbf{q}_d(t)$  oznacza błąd śledzenia położenia, a diagonalne macierze  $\mathbf{K}_d > 0$  oraz  $\mathbf{K}_p > 0$  opisują wzmocnienia części różniczkującej oraz proporcjonalnej regulatora PD. Równanie błędu w zamkniętej pętli sprzężenia zwrotnego jest następujące

$$\ddot{\mathbf{e}} + \mathbf{K}_d\dot{\mathbf{e}} + \mathbf{K}_p\mathbf{e} = \mathbf{0},$$

a zaproponowany wybór macierzy  $\mathbf{K}_d$  i  $\mathbf{K}_p$  zapewnia eksponencjalną zbieżność błędu śledzenia do zera.



**Rysunek 6.1** Przebieg błędów śledzenia przy zastosowaniu algorytmu linearyzacji przez sprzężenie zwrotne ze wzmocnieniami  $\mathbf{K}_p = \text{diag}\{300\}$  i  $\mathbf{K}_d = \text{diag}\{50\}$ .

### Przykład 6.1.1 (Manipulator *EDDA*)

Założmy, że zadanie sterowania manipulatora *EDDA* o modelu dynamiki zdefiniowanym przez (5.16), (5.21)–(5.23) polega na śledzeniu trajektorii zadanej w przestrzeni przegubowej

$$\mathbf{q}_d(t) = \begin{pmatrix} q_{1d}(t) \\ q_{2d}(t) \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{1}{2}t\right) \\ \sin(2t) \end{pmatrix}. \quad (6.7)$$

Błędy śledzenia powyższej trajektorii manipulatora zostały przedstawione na rysunku 6.1\*.

### Algorytm Wena-Bayarda

Drugim przedstawicielem algorytmów typu obliczanego momentu jest algorytm zaproponowany przez Wena i Bayarda

$$\mathbf{u} = \mathbf{Q}(\mathbf{q}_d)\ddot{\mathbf{q}}_d + \mathbf{C}(\mathbf{q}_d, \dot{\mathbf{q}}_d)\dot{\mathbf{q}}_d + \mathbf{D}(\mathbf{q}_d) - \mathbf{K}_d\dot{\mathbf{e}} - \mathbf{K}_p\mathbf{e}. \quad (6.8)$$

Łatwo zauważyć, że algorytm (6.8) składa się z dwóch składników. Pierwszy z nich, o charakterze korekcyjnym, wymaga znajomości modelu manipulatora wzdłuż zadanej trajektorii przegubowej. Drugi składnik, za pomocą regulatora PD o stałym wzmocnieniu, generuje poprawkę zależną od błędu śledzenia  $\mathbf{e}(t) = \mathbf{q}(t) - \mathbf{q}_d(t)$  i jego pochodnej względem czasu. W celu wykazania zbieżności algorytmu sterowania manipulatora opisanego równaniem (6.8) rozważmy funkcję Lapunowa

$$V(\mathbf{e}, \dot{\mathbf{e}}, t) = \frac{1}{2}\dot{\mathbf{e}}^T\mathbf{Q}(\mathbf{q})\dot{\mathbf{e}} + \frac{1}{2}\mathbf{e}^T(\mathbf{K}_p + \varepsilon\mathbf{K}_d)\mathbf{e} + \varepsilon\mathbf{e}^T\mathbf{Q}(\mathbf{q})\dot{\mathbf{e}}, \quad (6.9)$$

\*Symulacje zamieszczone w tym i następnym rozdziale wykonano w środowisku MATLAB® + SIMULINK®.

gdzie  $\mathbf{q} = \mathbf{q}_d + \mathbf{e}$ , a  $\varepsilon$  jest pewną stałą dodatnią. Z własności form kwadratowych<sup>†</sup> otrzymujemy warunek na  $\varepsilon$  zapewniający dodatnią określoność funkcji Lapunowa

$$0 < \varepsilon \leq \frac{\lambda_{\mathbf{K}_d} + \sqrt{\lambda_{\mathbf{K}_d}^2 + 4\Lambda_{\mathbf{Q}}\lambda_{\mathbf{K}_p}}}{2\Lambda_{\mathbf{Q}}},$$

przy czym  $\lambda_{\mathbf{Q}} = \min_{\mathbf{q}} \Lambda_{\mathbf{Q}}(\mathbf{q})$ ,  $\Lambda_{\mathbf{Q}} = \max_{\mathbf{q}} \Lambda_{\mathbf{Q}}(\mathbf{q})$ , zaś  $\lambda_{\mathbf{M}}$  oznacza najmniejszą wartość własną symetrycznej macierzy  $\mathbf{M}$ . Zauważmy, że istnienie  $\Lambda_{\mathbf{Q}}$  wynika z własności (5.17). Przed oszacowaniem pochodnej funkcji Lapunowa  $V$  zapiszmy równania manipulatora z zamkniętą pętlą sprzężenia zwrotnego

$$\begin{cases} \mathbf{Q}(\mathbf{q})\ddot{\mathbf{e}} = -\Delta\mathbf{Q}\ddot{\mathbf{q}}_d - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{e}} - \Delta\mathbf{C}\dot{\mathbf{q}}_d - \Delta\mathbf{D} - \mathbf{K}_p\mathbf{e} - \mathbf{K}_d\dot{\mathbf{e}} \\ \mathbf{q} = \mathbf{q}_d + \mathbf{e} \\ \dot{\mathbf{q}} = \dot{\mathbf{q}}_d + \dot{\mathbf{e}}, \end{cases} \quad (6.10)$$

gdzie  $\Delta\mathbf{Q} = \mathbf{Q}(\mathbf{q}) - \mathbf{Q}(\mathbf{q}_d)$ ,  $\Delta\mathbf{C} = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{C}(\mathbf{q}_d, \dot{\mathbf{q}}_d)$ ,  $\Delta\mathbf{D} = \mathbf{D}(\mathbf{q}) - \mathbf{D}(\mathbf{q}_d)$ . Korzystając z twierdzenia o wartości średniej<sup>‡</sup> można pokazać, że pochodna po czasie funkcji Lapunowa opisanej wzorem (6.9) wzdłuż trajektorii układu zamkniętego (6.10) da się oszacować w następujący sposób

$$\dot{V} \leq -B\|\mathbf{e}\|^2 - (A - S\|\mathbf{e}\|)\|\dot{\mathbf{e}}\|^2,$$

gdzie  $A, B, S$  są pewnymi liczbami dodatnimi. Wartość parametru  $S$  wynika z oszacowania modelu dynamiki wzdłuż trajektorii zadanej i dla konkretnej trajektorii zadanej jest ustalona. Parametr  $B$  można dowolnie powiększać poprzez wybór coraz większych wartości wzmocnień  $\mathbf{K}_p$ , natomiast parametr  $A$  rośnie wraz ze wzrostem wzmocnień  $\mathbf{K}_d$ .

Zdefiniujmy parametr  $\xi_1 = \frac{1}{2}(\lambda_{\mathbf{K}_p} + \varepsilon\lambda_{\mathbf{K}_d} - \varepsilon^2\Lambda_{\mathbf{Q}})$ . Jeśli zachodzi

$$B > 0, \quad A > S\sqrt{\frac{V_0}{\xi_1}},$$

gdzie  $V_0 = V(\mathbf{e}(0), \dot{\mathbf{e}}(0), 0)$ , to można udowodnić, że dla  $\lambda_1, \lambda_2$  spełniających

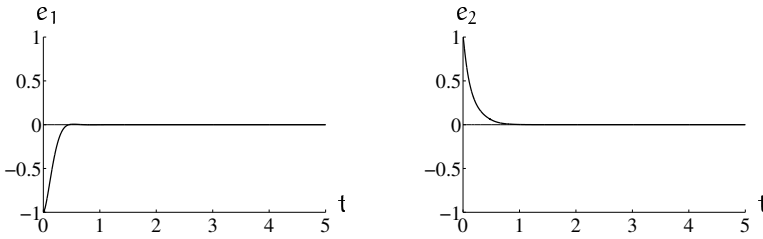
$$0 < \lambda_1 < B, \quad 0 < \lambda_2 < A - S\sqrt{\frac{V_0}{\xi_1}},$$

jest prawdziwa nierówność

$$\dot{V} \leq -\lambda_1\|\mathbf{e}\|^2 - \lambda_2\|\dot{\mathbf{e}}\|^2. \quad (6.11)$$

<sup>†</sup>Zobacz dodatek A.1.

<sup>‡</sup>Zobacz dodatek A.2.



**Rysunek 6.2** Przebieg błędów śledzenia przy zastosowaniu algorytmu Wena-Bayarda ze wzmocnieniami  $\mathbf{K}_p = \text{diag}\{300\}$  i  $\mathbf{K}_d = \text{diag}\{50\}$ .

Algorytm Wena-Bayarda zapewnia tzw. półglobalną stabilność zamkniętego układu sterowania. Oznacza to, że dla każdego warunku początkowego istnieją wzmocnienia  $\mathbf{K}_d$  i  $\mathbf{K}_p$  zapewniające eksponencjalną zbieżność błędu śledzenia do zera<sup>§</sup>.

### Przykład 6.1.2 (Manipulator *EDDA*)

Podobnie jak w algorytmie linearyzacji statycznej, celem sterowania jest śledzenie trajektorii przegubowej opisanej równaniem (6.7). Błędy powstałe przy zastosowaniu algorytmu Wena-Bayarda do śledzenia trajektorii zadanej przedstawiono na rysunku 6.2. ■

## 6.1.2 Algorytmy typu dysypatywnego

### Algorytm Slotine'a-Li

Najbardziej znanym algorytmem sterowania manipulatorów typu dysypatywnego jest algorytm Slotine'a-Li

$$\mathbf{u} = \mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}}_r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}_r + \mathbf{D}(\mathbf{q}) - \mathbf{K}_d\mathbf{s}, \quad (6.12)$$

w którym funkcja  $\mathbf{q}_r(t)$  jest tzw. trajektorią odniesienia zdefiniowaną następująco

$$\dot{\mathbf{q}}_r = \dot{\mathbf{q}}_d - \boldsymbol{\Lambda}\mathbf{e}, \quad (6.13)$$

natomiast zmienna

$$\mathbf{s} = \dot{\mathbf{q}} - \dot{\mathbf{q}}_r = \dot{\mathbf{e}} + \boldsymbol{\Lambda}\mathbf{e}, \quad (6.14)$$

nosi nazwę zmiennej ślizgu.

<sup>§</sup>Zauważmy, że im większy jest początkowy błąd śledzenia, tym wzmocnienia te muszą być większe.

Diagonalne macierze  $\mathbf{\Lambda} > 0$  i  $\mathbf{K}_d > 0$  są parametrami algorytmu sterowania. Łatwo pokazać, że równania manipulatora z zamkniętą pętlą sprzężenia zwrotnego są następujące

$$\begin{cases} \mathbf{Q}(\mathbf{q})\dot{\mathbf{s}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{s} + \mathbf{K}_d\mathbf{s} = \mathbf{0} \\ \mathbf{q} = \mathbf{e} + \mathbf{q}_d \\ \dot{\mathbf{e}} = \mathbf{s} - \mathbf{\Lambda}\mathbf{e}. \end{cases} \quad (6.15)$$

Dla dowodu stabilności algorytmu Slotine'a-Li można wykorzystać następującą funkcję Lapunowa

$$V(\mathbf{e}, \mathbf{s}, t) = \frac{1}{2}\mathbf{s}^T\mathbf{Q}(\mathbf{q})\mathbf{s}. \quad (6.16)$$

Pochodna względem czasu funkcji  $V(\mathbf{e}, \mathbf{s}, t)$  jest równa

$$\dot{V} = \mathbf{s}^T\mathbf{Q}(\mathbf{q})\dot{\mathbf{s}} + \frac{1}{2}\mathbf{s}^T\dot{\mathbf{Q}}(\mathbf{q})\mathbf{s}.$$

Policzymy pochodną funkcji Lapunowa wzdłuż trajektorii układu zamkniętego (6.15)

$$\begin{aligned} \dot{V} &= \mathbf{s}^T(-\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{s} - \mathbf{K}_d\mathbf{s}) + \frac{1}{2}\mathbf{s}^T\dot{\mathbf{Q}}(\mathbf{q})\mathbf{s} = \\ &= -\mathbf{s}^T\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{s} - \mathbf{s}^T\mathbf{K}_d\mathbf{s} + \frac{1}{2}\mathbf{s}^T\dot{\mathbf{Q}}(\mathbf{q})\mathbf{s}. \end{aligned}$$

Z własności (5.6) i z własności form kwadratowych otrzymujemy

$$\dot{V} = -\mathbf{s}^T\mathbf{K}_d\mathbf{s} \leq -\underline{\lambda}_{\mathbf{K}_d}\|\mathbf{s}\|^2. \quad (6.17)$$

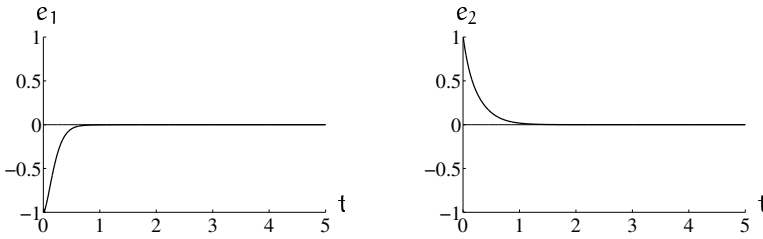
Korzystając z oszacowania

$$V(\mathbf{e}, \mathbf{s}, t) \leq \frac{1}{2}\bar{\lambda}_{\mathbf{Q}}\|\mathbf{s}\|^2$$

i z nierówności (6.17) można pokazać eksponencjalną zbieżność algorytmu\*

$$\mathbf{s} \rightarrow \mathbf{0}, \quad \text{gdy} \quad t \rightarrow +\infty.$$

Ponieważ  $\mathbf{s} = \dot{\mathbf{e}} + \mathbf{\Lambda}\mathbf{e} \rightarrow \mathbf{0}$ , to przy odpowiednim wyborze elementów macierzy  $\mathbf{\Lambda}$  błąd śledzenia  $\mathbf{e}(t)$  również zanika do zera eksponencjalnie.



**Rysunek 6.3** Przebieg błędów śledzenia przy algorytmie Slotine'a-Li z parametrami  $\mathbf{A} = \text{diag}\{4\}$  i  $\mathbf{K}_d = \text{diag}\{75\}$ .

### Przykład 6.1.3 (Manipulator *EDDA*)

W celu zilustrowania zachowania manipulatora przy sterowaniu według algorytmu Slotine'a-Li, przeprowadziliśmy symulacje komputerowe przebiegu śledzenia trajektorii przegubowej (6.7) manipulatora *EDDA*. Wykresy błędów zostały przedstawione na rysunku 6.3. ■

Zwróćmy uwagę na fakt, że składnik  $\mathbf{K}_d \mathbf{s}$  algorytmu Slotine'a-Li tłumiący błędy jest równoważny regulatorowi PD o wzmocnieniach  $\mathbf{K}_d \mathbf{A}$  i  $\mathbf{K}_d$ . Widać, że elementy macierzy  $\mathbf{A}$ , będąc składnikiem wzmocnienia części proporcjonalnej P regulatora, wpływają na szybkość dążenia błędu śledzenia trajektorii  $\mathbf{e}(t)$  do zera.

### Algorytm Sadegha-Horowitza

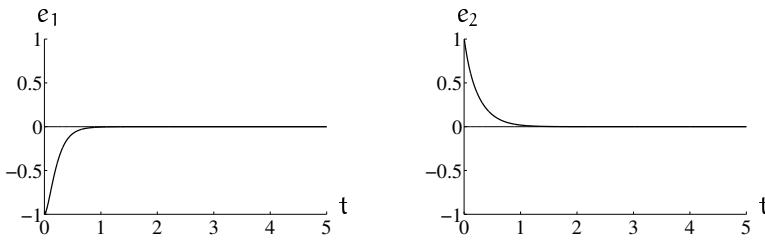
Innym algorytmem wykorzystującym pojęcie ślizgu jest algorytm sterowania Sadegha-Horowitza

$$\mathbf{u} = \mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}}_r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}_r + \mathbf{D}(\mathbf{q}) - \mathbf{K}_d \mathbf{s} - \mathbf{R}\mathbf{e}, \quad (6.18)$$

w którym wszystkie zmienne zostały zdefiniowane tak samo jak w algorytmie Slotine'a-Li. Jedynym elementem różniącym oba wspomniane algorytmy jest wyraz proporcjonalny do błędu  $\mathbf{R}\mathbf{e}$  występujący we wzorze (6.18), z diagonalną macierzą wzmocnień  $\mathbf{R} > 0$ .

Algorytm sterowania Sadegha-Horowitza jest pewną modyfikacją algorytmu Slotine'a-Li. Można pokazać za pomocą drugiej metody Lapunowa, że algorytm Sadegha-Horowitza jest eksponencjalnie stabilny. Dowód eksponencjalnej stabilności algorytmu Sadegha-Horowitza został umieszczony w dodatku C.

\*Zobacz dodatek C, wzór (C.9).



**Rysunek 6.4** Przebieg błędów śledzenia przy zastosowaniu algorytmu Sadegha-Horowitza z parametrami  $\mathbf{K}_d = \text{diag}\{75\}$ ,  $\mathbf{\Lambda} = \text{diag}\{2\}$  i  $\mathbf{R} = \text{diag}\{150\}$ .

### Przykład 6.1.4 (Manipulator EDDA)

Przykładowe przebiegi błędów śledzenia trajektorii przegubowej (6.7) manipulatora EDDA zostały zamieszczone na rysunku 6.4. ■

Analogicznie jak w przypadku algorytmu Slotine'a-Li, drugi składnik algorytmu Sadegha-Horowitza, równy  $\mathbf{K}_d \mathbf{s} + \mathbf{R} \mathbf{e}$ , jest równoważny regulatorowi PD o wzmocnieniach  $\mathbf{P} = \mathbf{K}_d \mathbf{\Lambda} + \mathbf{R}$  i  $\mathbf{D} = \mathbf{K}_d$ .

### Uniwersalny adaptacyjny algorytm sterowania

Znajomości modelu manipulatora wzdłuż zadanej trajektorii przegubowej wymaga *uniwersalny adaptacyjny algorytm sterowania* postaci

$$\mathbf{u} = \mathbf{Q}(\mathbf{q}_d) \ddot{\mathbf{q}}_d + \mathbf{C}(\mathbf{q}_d, \dot{\mathbf{q}}_d) \dot{\mathbf{q}}_d + \mathbf{D}(\mathbf{q}_d) - \mathbf{k}(t) \mathbf{P}_1 \mathbf{e} - \mathbf{k}(t) \mathbf{P}_2 \dot{\mathbf{e}}. \quad (6.19)$$

Stałe  $\mathbf{P}_1$ ,  $\mathbf{P}_2$  są dodatnie, natomiast  $\mathbf{e}(t)$  oznacza błąd śledzenia położenia zdefiniowany następująco

$$\mathbf{e}(t) = (e_1(t), \dots, e_i(t), \dots, e_n(t))^T = \mathbf{q}(t) - \mathbf{q}_d(t).$$

Macierz  $\mathbf{k}(t) = \text{diag}\{k_i(t)\}$ ,  $i = 1, \dots, n$ , reprezentuje wzmocnienia lokalnych uniwersalnych układów śledzenia. Lokalny układ uniwersalny dla  $i$ -tego przegubu manipulatora ma wzmocnienie zmieniane według reguły

$$\dot{k}_i(t) = (\mathbf{P}_1 e_i + \mathbf{P}_2 \dot{e}_i)^2, \quad i = 1, \dots, n. \quad (6.20)$$

Często przyjmuje się założenie, że  $k_i(0) > 0$ , co zapewnia dodatniość wzmocnień  $k_i(t)$  w każdej chwili czasu  $t \geq 0$ .

Z postaci równania (6.19) widać, że sterowanie manipulatora jest sumą dwóch składników

$$\mathbf{u} = \mathbf{u}_d + \mathbf{u}_{uniw}.$$

Składnik  $\mathbf{u}_d$  ma charakter korekcyjny i jest liczony w oparciu o znajomość modelu manipulatora wzdłuż zadanej trajektorii. Drugi składnik sterowania jest regulatorem PD o dynamicznym wzmocnieniu  $\mathbf{k}(t)$  generowanym przez układ uniwersalny

$$\mathbf{u}_{univ} = -\mathbf{k}(t)\mathbf{P}_1\mathbf{e} - \mathbf{k}(t)\mathbf{P}_2\dot{\mathbf{e}}. \quad (6.21)$$

Aby dowieść stabilności algorytmu sterowania, wybierzmy następującą funkcję Lapunowa

$$V(\mathbf{e}, \dot{\mathbf{e}}, \mathbf{k}, t) = \frac{1}{2}\dot{\mathbf{e}}^T\mathbf{Q}(\mathbf{q})\dot{\mathbf{e}} + \frac{1}{2}\mathbf{e}^T\mathbf{k}(\mathbf{P}_1 + \varepsilon\mathbf{P}_2)\mathbf{e} + \varepsilon\mathbf{e}^T\mathbf{Q}(\mathbf{q})\dot{\mathbf{e}}, \quad (6.22)$$

gdzie  $\mathbf{q} = \mathbf{q}_d + \mathbf{e}$ ,  $\varepsilon > 0$ , a  $\mathbf{k} = \mathbf{k}(t)$  jest macierzą dynamicznych wzmocnień lokalnych uniwersalnych układów śledzenia. Podobnie jak w algorytmie Wena-Bayarda, warunkiem dodatniej określoności funkcji Lapunowa jest

$$0 < \varepsilon \leq \frac{P_2\lambda_{\mathbf{k}(0)} + \sqrt{P_2^2\lambda_{\mathbf{k}(0)}^2 + 4\lambda_{\mathbf{Q}}P_1\lambda_{\mathbf{k}(0)}}}{2\lambda_{\mathbf{Q}}}.$$

Przed przystąpieniem do oszacowania pochodnej funkcji Lapunowa  $\dot{V}$ , zapiszmy równania manipulatora z zamkniętą pętlą sprzężenia zwrotnego (6.19)

$$\begin{cases} \mathbf{Q}(\mathbf{q})\ddot{\mathbf{e}} = -\Delta\mathbf{Q}\ddot{\mathbf{q}}_d - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{e}} - \Delta\mathbf{C}\dot{\mathbf{q}}_d - \Delta\mathbf{D} - \mathbf{k}\mathbf{P}_1\mathbf{e} - \mathbf{k}\mathbf{P}_2\dot{\mathbf{e}} \\ \dot{\mathbf{k}} = \text{diag}\{P_1\dot{e}_i + P_2\dot{e}_i\}^2 \\ \mathbf{q} = \mathbf{q}_d + \mathbf{e} \\ \dot{\mathbf{q}} = \dot{\mathbf{q}}_d + \dot{\mathbf{e}}, \end{cases} \quad (6.23)$$

przy oznaczeniach  $\Delta\mathbf{Q} = \mathbf{Q}(\mathbf{q}) - \mathbf{Q}(\mathbf{q}_d)$ ,  $\Delta\mathbf{C} = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{C}(\mathbf{q}_d, \dot{\mathbf{q}}_d)$ ,  $\Delta\mathbf{D} = \mathbf{D}(\mathbf{q}) - \mathbf{D}(\mathbf{q}_d)$ . Twierdzenie o wartości średniej pozwala pokazać, że pochodna względem czasu funkcji Lapunowa opisanej wzorem (6.22) wzdłuż trajektorii układu zamkniętego (6.23) da się oszacować w następujący sposób

$$\dot{V} \leq -(B - E\|\mathbf{e}\|^2 - D\|\dot{\mathbf{e}}\|)\|\mathbf{e}\|^2 - (A - S\|\mathbf{e}\|)\|\dot{\mathbf{e}}\|^2.$$

Parametry A, B, D, E, S są pewnymi liczbami dodatnimi. Wartość parametru S wynika z oszacowania modelu dynamiki wzdłuż trajektorii zadanej i dla konkretnej trajektorii zadanej jest ustalona. Parametry A, B, D, E można dowolnie powiększać poprzez odpowiedni wybór wartości  $P_1$  i  $P_2$ .



Zdefiniujmy wyrażenia

$$\xi_1 = \frac{1}{2} \left( P_1 \underline{\lambda}_{\mathbf{k}(0)} + \varepsilon P_2 \underline{\lambda}_{\mathbf{k}(0)} - \varepsilon^2 \Lambda_{\mathbf{Q}} \right),$$

$$\xi_2 = \frac{\underline{\lambda}_{\mathbf{Q}} \xi_1}{P_1 \underline{\lambda}_{\mathbf{k}(0)} + \varepsilon P_2 \underline{\lambda}_{\mathbf{k}(0)}}.$$

Jeśli spełnione są nierówności

$$B > E \frac{V_0}{\xi_1} + D \sqrt{\frac{V_0}{\xi_2}}, \quad A > S \sqrt{\frac{V_0}{\xi_1}},$$

gdzie  $V_0 = V(\mathbf{e}(0), \dot{\mathbf{e}}(0), 0)$ ,  $\Lambda_{\mathbf{Q}} = \max_{\mathbf{q}} \Lambda_{\mathbf{Q}}(\mathbf{q})$  oraz  $\underline{\lambda}_{\mathbf{Q}} = \min_{\mathbf{q}} \Lambda_{\mathbf{Q}}(\mathbf{q})$ , to można pokazać, że dla  $\lambda_1, \lambda_2$  spełniających

$$0 < \lambda_1 < B - E \frac{V_0}{\xi_1} - D \sqrt{\frac{V_0}{\xi_2}}, \quad 0 < \lambda_2 < A - S \sqrt{\frac{V_0}{\xi_1}},$$

prawdziwa jest nierówność

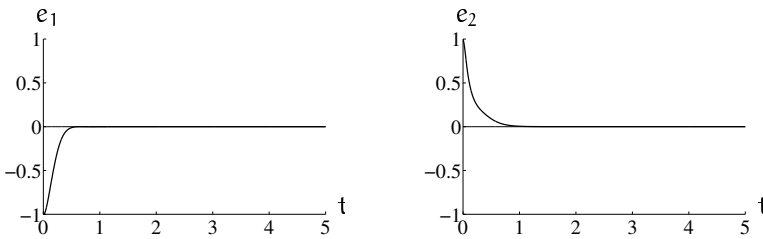
$$\dot{V} \leq -\lambda_1 \|\mathbf{e}\|^2 - \lambda_2 \|\dot{\mathbf{e}}\|^2. \quad (6.24)$$

Przedstawiony uniwersalny algorytm sterowania jest półglobalnie asymptotycznie stabilny, przy czym błąd śledzenia trajektorii  $\mathbf{e}(t)$  oraz błąd śledzenia prędkości  $\dot{\mathbf{e}}(t)$  zanikają do zera eksponencjalnie. Zaletą algorytmu uniwersalnego w porównaniu z algorytmami o statycznym wzmocnieniu w pętli sprzężenia zwrotnego jest mniejsza wartość sił i momentów sterujących  $\mathbf{u}$  potrzebnych do uzyskania pożądanego zachowania manipulatora.

#### Przykład 6.1.5 (Manipulator *EDDA*)

Przykładowe przebiegi błędów śledzenia trajektorii przegubowej (6.7) manipulatora *EDDA* przy wykorzystaniu uniwersalnego adaptacyjnego algorytmu sterowania zostały zamieszczone na rysunku 6.5. W trakcie symulacji wzmocnienia lokalnych układów uniwersalnych ustaliły się na wartościach  $k_{1ust} = 17$  i  $k_{2ust} = 6.5$ . ■

Symulacje pokazują, że wszystkie algorytmy przedstawione w podrozdziałach 6.1.1 i 6.1.2 zapewniają porównywalną jakość śledzenia trajektorii, natomiast różnią się pod względem łatwości implementacji komputerowej. W praktyce najłatwiej jest zaimplementować algorytm Wena-Bayarda.



**Rysunek 6.5** Przebieg błędów śledzenia przy zastosowaniu uniwersalnego adaptacyjnego algorytmu sterowania z parametrami  $P_1 = 20$ ,  $P_2 = 4$  i  $k_i(0) = 0.1$ .

W tym algorytmie składowa korekcyjna  $\mathbf{u}_d$  sterowania może zostać obliczona w trybie *off-line*, co zmniejsza zarówno zajętość pamięci komputera, jak i czas trwania bieżących obliczeń. Uniwersalny adaptacyjny algorytm sterowania może być traktowany jako dynamiczna wersja algorytmu Wena-Bayarda. Składową  $\mathbf{u}_d$  sterowania również wylicza się w tym przypadku *off-line*, jednak konieczność wyznaczania dynamicznego wzmocnienia zwiększa wymagania co do wielkości potrzebnych zasobów obliczeniowych. Pozostałe algorytmy wymagają obliczania dodatkowych zmiennych, a przez to są jeszcze trudniejsze do implementacji.

## 6.2 Algorytmy sterowania przy parametrycznej nieznajomości modelu

Jak powiedzieliśmy na wstępie, nieznajomością parametryczną modelu nazywa się sytuację, gdy znane są wszystkie elementy modelu dynamiki manipulatora (znana jest postać funkcyjna wszystkich zależności między zmiennymi opisującymi zachowanie), natomiast nie są znane parametry występujące w tych równaniach. Z postaci modelu dynamiki manipulatora przedstawionej w podrozdziale 5.2 wynika, że każde ramię jest opisane przy pomocy dziesięciu parametrów dynamiki (elementów macierzy inercji ramienia), zatem model manipulatora o  $n$  stopniach swobody zawiera  $10n$  parametrów dynamiki, które zależą od jego własności fizycznych. Na ogół parametry dynamiki są nieznane lub znane tylko w przybliżeniu. Najczęstszą przyczyną zmiany tych parametrów jest zmienna masa i moment bezwładności ładunku przeniesionego przez manipulator. W dalszych rozważaniach przyjmujemy założenie, że w trakcie śledzenia konkretnej trajektorii przegubowej parametry dynamiki pozostają ustalone, aczkolwiek mogą być nieznane.

W przedstawionych niżej adaptacyjnych algorytmach sterowania będziemy posługiwać modelem manipulatora, który jest liniowy ze względu na nieznanne parametry  $\theta$

$$\mathbf{Q}(\mathbf{q}, \theta)\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \theta)\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}, \theta) = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\theta = \mathbf{u}, \quad (6.25)$$

gdzie  $\theta \in \mathbb{R}^p$ , zaś  $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$  jest macierzą rozmiaru  $n \times p$  zwaną *macierzą regresji*. Parametry  $\theta$  występujące w równaniu (6.25) są parametrami zastępczymi modelu, będącymi pewnymi funkcjami parametrów fizycznych, a nie bezpośrednio parametrami fizycznymi. W zapisie macierzy regresji przyjmujemy, że trzeci argument odnosi się do prędkości  $\dot{\mathbf{q}}$ , która jest mnożona przez macierz  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ . Nietrudno zauważyć, że macierz regresji  $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$  zależy w sposób liniowy od trzeciego i czwartego argumentu.

### Przykład 6.2.1 (Manipulator *EDDA*)

Manipulator *EDDA* opisany równaniami macierzowo-wektorowymi (5.16) z macierzami  $\mathbf{Q}$ ,  $\mathbf{C}$  i wektorem  $\mathbf{D}$  zdefiniowanymi przez (5.21)–(5.23) posiada następujący równoważny opis w postaci (6.25):

$$\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\theta = \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} & Y_{14} \\ 0 & 0 & Y_{23} & Y_{24} \end{bmatrix} \theta. \quad (6.26)$$

Elementy  $Y_{ij}$  macierzy w równaniu (6.26) są następujące:

$$\begin{aligned} Y_{11} &= \ddot{q}_1, \\ Y_{12} &= g \cos q_1, \\ Y_{13} &= \ddot{q}_2, \\ Y_{14} &= 0.3(2\ddot{q}_1 + \ddot{q}_2) \cos q_2 + g \cos(q_1 + q_2) + \\ &\quad - 0.3(\dot{q}_1^2 + \dot{q}_1 \dot{q}_2 + \dot{q}_2^2) \sin q_2, \\ Y_{23} &= \ddot{q}_1 + \ddot{q}_2, \\ Y_{24} &= 0.3\ddot{q}_1 \cos q_2 + g \cos(q_1 + q_2) + 0.3\dot{q}_1^2 \sin q_2, \end{aligned}$$

natomiast wektor  $\theta = (\theta_1, \dots, \theta_4)^T$  jest wektorem nieznanych parametrów zastępczych modelu, których wartości nominalne podaje formuła (5.24). Model manipulatora *EDDA* opisany równaniami (6.26) zostanie użyty do symulacji komputerowych adaptacyjnych algorytmów sterowania. ■

Przy projektowaniu adaptacyjnego algorytmu sterowania zapewniającego śledzenie zadanej trajektorii  $\mathbf{q}_d(t)$  w układzie (6.25) należy postępować

w następujący sposób:

- założyć, że w każdej chwili dysponujemy pewną oceną, czyli estymatą  $\hat{\boldsymbol{\theta}}(t)$  wektora nieznanymi parametrów  $\boldsymbol{\theta}$ ,
- znaleźć algorytm sterowania  $\mathbf{u}$  wykorzystujący bieżącą wartość estymaty wektora nieznanymi parametrów,
- znaleźć algorytm estymacji nieznanymi parametrów generujący w każdej chwili kolejną ocenę  $\hat{\boldsymbol{\theta}}(t)$ ,
- udowodnić asymptotyczną stabilność adaptacyjnego układu sterowania złożonego z podsystemu sterowania i podsystemu estymacji.

### 6.2.1 Algorytmy adaptacyjne typu obliczanego momentu

#### Adaptacyjny algorytm linearyzacji

*Adaptacyjny algorytm linearyzacji* otrzymuje się wprost z algorytmu sterowania zaprojektowanego dla przypadku nieadaptacyjnego (6.4). W tym celu, zamiast rzeczywistych wartości nieznanymi parametrów, należy wykorzystać ich bieżące estymaty

$$\begin{cases} \mathbf{u} = \mathbf{Q}(\mathbf{q}, \hat{\boldsymbol{\theta}}) \mathbf{v} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \hat{\boldsymbol{\theta}}) \dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}, \hat{\boldsymbol{\theta}}) \\ \mathbf{v} = \ddot{\mathbf{q}}_d - \mathbf{K}_d \dot{\mathbf{e}} - \mathbf{K}_p \mathbf{e}. \end{cases} \quad (6.27)$$

Wstawiając sterowanie (6.27) do modelu dynamiki manipulatora (6.25) otrzymujemy równanie układu z zamkniętą pętlą sprzężenia zwrotnego

$$\mathbf{Q}(\mathbf{q}, \hat{\boldsymbol{\theta}}) (\ddot{\mathbf{e}} + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e}) = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \hat{\boldsymbol{\theta}}) \tilde{\boldsymbol{\theta}}, \quad (6.28)$$

w którym  $\tilde{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t) - \boldsymbol{\theta}$  jest błędem estymacji parametrów. Przy założeniu, że macierz  $\mathbf{Q}(\mathbf{q}, \hat{\boldsymbol{\theta}})$  jest nieosobliwa, równanie (6.28) przyjmuje postać

$$\ddot{\mathbf{e}} + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e} = \mathbf{Q}^{-1}(\mathbf{q}, \hat{\boldsymbol{\theta}}) \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \hat{\boldsymbol{\theta}}) \tilde{\boldsymbol{\theta}},$$

lub równoważną postać wyrażoną za pomocą zmiennych stanu

$$\dot{\mathbf{z}} = \mathbf{A} \mathbf{z} + \mathbf{B} \mathbf{Q}^{-1}(\mathbf{q}, \hat{\boldsymbol{\theta}}) \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \hat{\boldsymbol{\theta}}) \tilde{\boldsymbol{\theta}}, \quad \mathbf{z} = (\mathbf{e}^T, \dot{\mathbf{e}}^T)^T, \quad (6.29)$$

gdzie

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbb{I}_n \\ -\mathbf{K}_p & -\mathbf{K}_d \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbb{I}_n \end{bmatrix}.$$

Algorytm estymacji parametrów jest następujący

$$\dot{\hat{\boldsymbol{\theta}}}(t) = \dot{\boldsymbol{\theta}}(t) = -\boldsymbol{\Gamma}\mathbf{Y}^T(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) (\mathbf{Q}^T)^{-1}(\mathbf{q}, \hat{\boldsymbol{\theta}}) \mathbf{B}^T \mathbf{P}\mathbf{z}. \quad (6.30)$$

Macierz  $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}^T > 0$  jest macierzą wzmocnień algorytmu estymacji, natomiast macierz  $\mathbf{P} = \mathbf{P}^T > 0$  jest rozwiązaniem równania Lapunowa

$$\mathbf{A}^T \mathbf{P} + \mathbf{P}\mathbf{A} = -\mathbf{M}, \quad (6.31)$$

dla pewnej macierzy  $\mathbf{M} = \mathbf{M}^T > 0$ , która będzie odpowiadać za szybkość zbieżności zmiennej  $\mathbf{z}$  do zera.

Wyberzmy następującą funkcję Lapunowa dla układu (6.29)–(6.30)

$$V(\mathbf{z}, \tilde{\boldsymbol{\theta}}) = \frac{1}{2} \mathbf{z}^T \mathbf{P}\mathbf{z} + \frac{1}{2} \tilde{\boldsymbol{\theta}}^T \boldsymbol{\Gamma}^{-1} \tilde{\boldsymbol{\theta}}. \quad (6.32)$$

Pochodna względem czasu funkcji  $V(\mathbf{z}, \tilde{\boldsymbol{\theta}}, t)$ , wyliczona wzdłuż trajektorii układu zamkniętego (6.29)–(6.30), jest równa

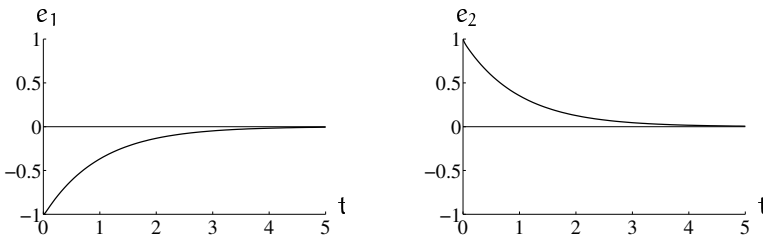
$$\dot{V}(\mathbf{z}, \tilde{\boldsymbol{\theta}}) = \frac{1}{2} \dot{\mathbf{z}}^T \mathbf{P}\mathbf{z} + \frac{1}{2} \mathbf{z}^T \dot{\mathbf{P}}\mathbf{z} + \tilde{\boldsymbol{\theta}}^T \boldsymbol{\Gamma}^{-1} \dot{\tilde{\boldsymbol{\theta}}} = -\frac{1}{2} \mathbf{z}^T \mathbf{M}\mathbf{z} \quad (6.33)$$

i zawiera macierz  $\mathbf{M}$  wprowadzoną w równaniu (6.31). Z równości (6.33) wynika, że adaptacyjny algorytm linearyzacji jest globalnie asymptotycznie stabilny.

### Przykład 6.2.2 (Manipulator *EDDA*)

Aby zilustrować działanie adaptacyjnego algorytmu linearyzacji, ponownie rozważymy zadanie śledzenia trajektorii (6.7) manipulatora *EDDA*. Wykresy błędów śledzenia w poszczególnych przegubach manipulatora zostały przedstawione na rysunku 6.6. ■

Prezentowany algorytm jest uogólnieniem algorytmu linearyzacji przez statyczne sprzężenie zwrotne na przypadek parametrycznej nieznajomości modelu dynamiki manipulatora. Algorytm posiada kilka własności utrudniających jego implementację. Po pierwsze, algorytm estymacji wymaga pomiaru przyspieszeń w przegubach. W praktyce jest to bardzo utrudnione, ponieważ większość robotów nie posiada czujników przyspieszenia, a wyliczanie sygnału przyspieszenia poprzez różniczkowanie sygnału położenia wprowadza znaczne zaszumienie sygnału przyspieszenia. Drugą istotną wadą



**Rysunek 6.6** Przebieg błędów śledzenia przy zastosowaniu adaptacyjnego algorytmu linearyzacji z parametrami  $\mathbf{K}_p = \text{diag}\{300\}$ ,  $\mathbf{K}_d = \text{diag}\{50\}$ ,  $\mathbf{\Gamma} = \text{diag}\{20\}$ .

adaptacyjnego algorytmu linearyzacji jest konieczność odwracania estymowanej macierzy bezwładności  $\mathbf{Q}(\mathbf{q}, \hat{\boldsymbol{\theta}})$ . Odwracanie macierzy  $\mathbf{Q}(\mathbf{q}, \hat{\boldsymbol{\theta}})$  dla manipulatorów o wielu stopniach swobody jest nie tylko czasochłonne i skomplikowane numerycznie ale również często niemożliwe. Wynika stąd, że chociaż macierz bezwładności manipulatora z parametrami nominalnymi jest zawsze dodatnio określona, to estymowana macierz  $\mathbf{Q}(\mathbf{q}, \hat{\boldsymbol{\theta}})$  nie posiada tej własności i przy pewnych wartościach  $\hat{\boldsymbol{\theta}}$  może stać się osobliwa. Należy również podkreślić fakt, że algorytm estymacji nie gwarantuje zbieżności estymat parametrów dynamiki do ich rzeczywistych wartości.

### Algorytm adaptacyjny Bayarda-Wena

Bayard i Wen zaproponowali rodzinę adaptacyjnych algorytmów sterowania wykorzystujących stopniowo coraz mniejszą wiedzę o modelu dynamiki manipulatora. W obrębie tej rodziny najmniejszej wiedzy o dynamice manipulatora wymaga następujący algorytm sterowania

$$\begin{aligned} \mathbf{u} &= \mathbf{Q}(\mathbf{q}_d, \hat{\boldsymbol{\theta}}) \ddot{\mathbf{q}}_d + \mathbf{C}(\mathbf{q}_d, \dot{\mathbf{q}}_d, \hat{\boldsymbol{\theta}}) \dot{\mathbf{q}}_d + \mathbf{D}(\mathbf{q}_d, \hat{\boldsymbol{\theta}}) - \mathbf{K}_d \dot{\mathbf{e}} - \mathbf{K}_p \mathbf{e} = \\ &= \mathbf{Y}(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d, \hat{\boldsymbol{\theta}}) - \mathbf{K}_d \dot{\mathbf{e}} - \mathbf{K}_p \mathbf{e}, \end{aligned} \quad (6.34)$$

z algorytmem estymacji parametrów

$$\dot{\hat{\boldsymbol{\theta}}}(t) = -\mathbf{\Gamma} \mathbf{Y}^T(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d) (\dot{\mathbf{e}} + \boldsymbol{\varepsilon} \mathbf{e}). \quad (6.35)$$

Zakłada się, że  $\mathbf{K}_d = \mathbf{K}_d^T > 0$ ,  $\mathbf{K}_p = \mathbf{K}_p^T > 0$  i  $\mathbf{\Gamma} = \mathbf{\Gamma}^T > 0$ . Przed przystąpieniem do badania stabilności algorytmu przedstawimy równania dynamiki

manipulatora z zamkniętą pętlą sprzężenia zwrotnego (6.34), (6.35)

$$\begin{cases} \mathbf{Q}(\mathbf{q})\ddot{\mathbf{e}} = -\Delta\mathbf{Q}\ddot{\mathbf{q}}_d - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{e}} - \Delta\mathbf{C}\dot{\mathbf{q}}_d - \Delta\mathbf{D} + \\ \quad - \mathbf{K}_p\mathbf{e} - \mathbf{K}_d\dot{\mathbf{e}} + \mathbf{Y}(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d)\ddot{\boldsymbol{\theta}} \\ \dot{\hat{\boldsymbol{\theta}}}(t) = -\Gamma\mathbf{Y}^T(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d)(\dot{\mathbf{e}} + \varepsilon\mathbf{e}) \\ \mathbf{q} = \mathbf{q}_d + \mathbf{e} \\ \dot{\mathbf{q}} = \dot{\mathbf{q}}_d + \dot{\mathbf{e}}, \end{cases} \quad (6.36)$$

gdzie  $\Delta\mathbf{Q} = \mathbf{Q}(\mathbf{q}) - \mathbf{Q}(\mathbf{q}_d)$ ,  $\Delta\mathbf{C} = \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{C}(\mathbf{q}_d, \dot{\mathbf{q}}_d)$ ,  $\Delta\mathbf{D} = \mathbf{D}(\mathbf{q}) - \mathbf{D}(\mathbf{q}_d)$ .

Dla układu opisanego równaniami (6.36) wybieramy funkcję Lapunowa

$$V(\mathbf{e}, \dot{\mathbf{e}}, \ddot{\boldsymbol{\theta}}, t) = V^0 + \frac{1}{2}\ddot{\boldsymbol{\theta}}^T\Gamma^{-1}\ddot{\boldsymbol{\theta}}, \quad (6.37)$$

zawierającą funkcję Lapunowa  $V^0$  dla nieadaptacyjnego algorytmu Wena-Bayarda zdefiniowaną wzorem (6.9). Można pokazać, że pochodna funkcji  $V$  obliczona wzdłuż trajektorii układu (6.36) wynosi

$$\dot{V} = \dot{V}^0 + \dot{\mathbf{e}}^T\mathbf{Y}(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d)\ddot{\boldsymbol{\theta}} + \varepsilon\mathbf{e}^T\mathbf{Y}(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d)\ddot{\boldsymbol{\theta}} + \ddot{\boldsymbol{\theta}}^T\Gamma^{-1}\ddot{\boldsymbol{\theta}}. \quad (6.38)$$

Widać, że po wstawieniu równania (6.35) opisującego algorytm estymacji parametrów do wzoru (6.38), prawdziwe jest oszacowanie

$$\dot{V} \leq -B\|\mathbf{e}\|^2 - (A - S\|\mathbf{e}\|)\|\dot{\mathbf{e}}\|^2,$$

z parametrami  $A$ ,  $B$ ,  $S$  identycznymi jak w przypadku algorytmu Wena-Bayarda. Oznacza to, że dla  $\lambda_1$ ,  $\lambda_2$  spełniających zależności

$$0 < \lambda_1 < B, \quad 0 < \lambda_2 < A - S\sqrt{\frac{V_0}{\xi_1}},$$

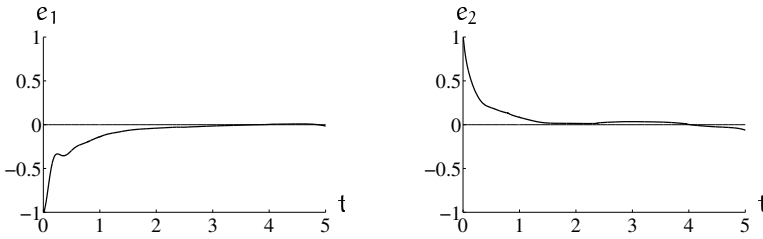
prawdziwa jest nierówność

$$\dot{V} \leq -\lambda_1\|\mathbf{e}\|^2 - \lambda_2\|\dot{\mathbf{e}}\|^2. \quad (6.39)$$

Wszystkie występujące tu zmienne i parametry zostały zdefiniowane w rozdziale omawiającym algorytm Wena-Bayarda\*. Algorytm (6.34), (6.35) jest

---

\*Kolejność nazwisk w algorytmach Wena-Bayarda i Bayarda-Wena nie jest przypadkowa, lecz wynika z kolejności nazwisk autorów artykułów, w których te algorytmy zostały opublikowane.



**Rysunek 6.7** Przebieg błędów śledzenia przy zastosowaniu adaptacyjnego algorytmu Bayarda-Wena z parametrami  $\mathbf{K}_p = \text{diag}\{300\}$ ,  $\mathbf{K}_d = \text{diag}\{50\}$  i  $\Gamma = \text{diag}\{20\}$ .

półglobalnie asymptotycznie stabilny. Oznacza to, że dla dowolnych warunków początkowych istnieją wzmocnienia regulatora PD (czyli wartości elementów macierzy  $\mathbf{K}_d$  i  $\mathbf{K}_p$ ), które zapewniają zbieżność błędu śledzenia trajektorii  $\mathbf{e}(t) = \mathbf{q}(t) - \mathbf{q}_d(t)$  do zera. Zaletą algorytmu Bayarda-Wena jest duża odporność na zakłócenia, natomiast istotną wadą są bardzo duże wartości wzmocnień  $\mathbf{K}_d$  i  $\mathbf{K}_p$  konieczne do zapewnienia stabilności algorytmu. Również ten algorytm nie gwarantuje zbieżności estymat do nominalnych wartości parametrów.

### Przykład 6.2.3 (Manipulator *EDDA*)

Aby zilustrować działanie powyższego algorytmu, pokażemy przykładowe wykresy błędów śledzenia trajektorii dla manipulatora typu *EDDA*. Podobnie jak w przypadku nieadaptacyjnym, trajektoria zadana jest opisana równaniem (6.7). Wykresy błędów śledzenia trajektorii przegubowej zostały przedstawione na rysunku 6.7. ■

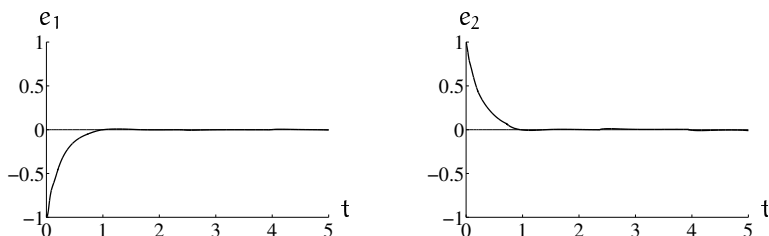
## 6.2.2 Algorytmy adaptacyjne typu dysypatywnego

### Algorytm adaptacyjny Slotine'a-Li

Adaptacyjny algorytm Slotine'a-Li wywodzi się z algorytmu (6.12) używanego przy pełnej znajomości modelu manipulatora, z tą różnicą, że zawarte w nim prawo sterowania zamiast rzeczywistych wartości wektora nieznanymi parametrów modelu  $\boldsymbol{\theta}$  wykorzystuje ich bieżące estymaty  $\hat{\boldsymbol{\theta}}$ :

$$\begin{aligned} \mathbf{u} &= \mathbf{Q}(\mathbf{q}, \hat{\boldsymbol{\theta}}) \ddot{\mathbf{q}}_r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \hat{\boldsymbol{\theta}}) \dot{\mathbf{q}}_r + \mathbf{D}(\mathbf{q}, \hat{\boldsymbol{\theta}}) - \mathbf{K}_d \mathbf{s} = \\ &= \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r) \hat{\boldsymbol{\theta}} - \mathbf{K}_d \mathbf{s}. \end{aligned} \quad (6.40)$$





**Rysunek 6.8** Przebieg błędów śledzenia przy zastosowaniu adaptacyjnego algorytmu Slotine'a-Li z parametrami  $\mathbf{K}_d = \text{diag}\{75\}$ ,  $\mathbf{\Lambda} = \text{diag}\{4\}$  i  $\mathbf{\Gamma} = \text{diag}\{20\}$ .

Wszystkie sygnały występujące we wzorze (6.40) są takie same jak sygnały dla algorytmu nieadaptacyjnego i zostały zdefiniowane następująco:

$$\begin{aligned} \mathbf{e} &= \mathbf{q} - \mathbf{q}_d, \\ \dot{\mathbf{q}}_r &= \dot{\mathbf{q}}_d - \mathbf{\Lambda}\mathbf{e}, \\ \mathbf{s} &= \dot{\mathbf{q}} - \dot{\mathbf{q}}_r = \dot{\mathbf{e}} + \mathbf{\Lambda}\mathbf{e}. \end{aligned} \quad (6.41)$$

Jako algorytm estymacji przyjmujemy następujący algorytm

$$\dot{\hat{\boldsymbol{\theta}}}(t) = -\mathbf{\Gamma}\mathbf{Y}^T(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)\mathbf{s}, \quad (6.42)$$

gdzie  $\mathbf{\Gamma}$  jest symetryczną, dodatnio określoną macierzą wzmocnień adaptacyjnych rozmiaru  $p \times p$ , zaś  $p$  jest liczbą parametrów zastępczych.

Adaptacyjny algorytm Slotine'a-Li cechuje się eksponencjalną zbieżnością błędu śledzenia trajektorii  $\mathbf{e}$  do zera. Dowód jego stabilności znajduje się w dodatku C. Algorytm ten nie gwarantuje zbieżności estymat nieznanymi parametrów do ich wartości nominalnych.

#### Przykład 6.2.4 (Manipulator *EDDA*)

W celu zaprezentowania działania powyższego algorytmu, na rysunku 6.8 przedstawimy wykresy błędów śledzenia trajektorii (6.7) dla manipulatora eksperymentalnego *EDDA*. ■

#### Algorytm adaptacyjny Sadegha-Horowitza

Podobnie jak Bayard i Wen, Sadegh i Horowitz zaproponowali rodzinę adaptacyjnych algorytmów sterowania wykorzystujących różny stopień znajomości modelu dynamiki manipulatora. Spośród adaptacyjnych algorytmów sterowania z tej rodziny rozważymy algorytm wykorzystujący prawo

sterowania

$$\begin{aligned} \mathbf{u} &= \mathbf{Q}(\mathbf{q}_d, \hat{\boldsymbol{\theta}}) \ddot{\mathbf{q}}_d + \mathbf{C}(\mathbf{q}_d, \dot{\mathbf{q}}_d, \hat{\boldsymbol{\theta}}) \dot{\mathbf{q}}_d + \mathbf{D}(\mathbf{q}_d, \hat{\boldsymbol{\theta}}) - \mathbf{K}_d \mathbf{s} - \mathbf{R} \mathbf{e} - K_f \|\mathbf{e}\|^2 \mathbf{s} = \\ &= \mathbf{Y}(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d, \hat{\boldsymbol{\theta}}) \hat{\boldsymbol{\theta}} - \mathbf{K}_d \mathbf{s} - \mathbf{R} \mathbf{e} - K_f \|\mathbf{e}\|^2 \mathbf{s} \end{aligned} \quad (6.43)$$

oraz algorytm estymacji parametrów

$$\dot{\hat{\boldsymbol{\theta}}}(t) = -\Gamma \mathbf{Y}^T(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d) \mathbf{s}. \quad (6.44)$$

W algorytmie  $\mathbf{e} = \mathbf{q} - \mathbf{q}_d$  jest błędem śledzenia trajektorii,  $\mathbf{s} = \dot{\mathbf{e}} + \boldsymbol{\Lambda} \mathbf{e}$  oznacza zmienną ślizgu, analogicznie do wersji nieadaptacyjnej, współczynnik  $K_f > 0$ , zaś macierze wzmocnień spełniają warunki  $\mathbf{K}_d = \mathbf{K}_d^T > \mathbf{0}$ ,  $\mathbf{R} = \mathbf{R}^T > \mathbf{0}$ ,  $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}^T > \mathbf{0}$  i  $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^T > \mathbf{0}$ . Do przeprowadzenia analizy stabilności powyższego algorytmu wykorzystamy równania dynamiki manipulatora z zamkniętą pętlą sprzężenia zwrotnego

$$\begin{cases} \mathbf{Q}(\mathbf{q}) \dot{\mathbf{s}} = -\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \mathbf{s} - \mathbf{R} \mathbf{e} - \mathbf{K}_d \mathbf{s} - K_f \|\mathbf{e}\|^2 \mathbf{s} + \mathbf{Y}(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d, \hat{\boldsymbol{\theta}}) \hat{\boldsymbol{\theta}} + \Delta \mathbf{Y} \\ \dot{\hat{\boldsymbol{\theta}}}(t) = -\Gamma \mathbf{Y}^T(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d) \mathbf{s} \\ \mathbf{q} = \mathbf{q}_d + \mathbf{e} \\ \mathbf{s} = \dot{\mathbf{q}} - \dot{\mathbf{q}}_r = \dot{\mathbf{e}} + \boldsymbol{\Lambda} \mathbf{e}, \end{cases} \quad (6.45)$$

gdzie  $\Delta \mathbf{Y} = (\mathbf{Y}(\mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d, \hat{\boldsymbol{\theta}}) - \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \hat{\boldsymbol{\theta}})) \hat{\boldsymbol{\theta}}$ . Dla układu (6.45) wybieramy funkcję Lapunowa

$$V(\mathbf{e}, \mathbf{s}, \hat{\boldsymbol{\theta}}, t) = V^0(\mathbf{e}, \mathbf{s}, t) + \frac{1}{2} \hat{\boldsymbol{\theta}}^T \boldsymbol{\Gamma}^{-1} \hat{\boldsymbol{\theta}} + \frac{1}{2} \lambda \underline{\lambda}^2 \mathbf{e}^T \mathbf{e}, \quad (6.46)$$

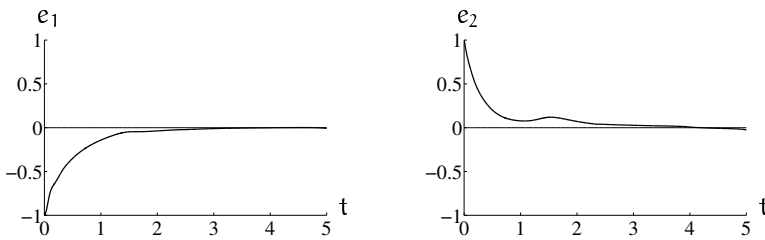
w której składnik  $V^0$  oznacza funkcję Lapunowa nieadaptacyjnego algorytmu Sadegha-Horowitza zdefiniowaną wzorem (C.4), a  $\lambda = \frac{1}{2} (\lambda_Q + \bar{\lambda}_Q)^*$ . Pochodną funkcji Lapunowa  $\dot{V}$  wzdłuż trajektorii układu (6.45) można oszacować następująco

$$\dot{V}(\mathbf{e}, \mathbf{s}, \hat{\boldsymbol{\theta}}, t) \leq -\sigma_1 \|\mathbf{s}\|^2 - \underline{\lambda} \sigma_2 \|\mathbf{e}\|^2, \quad (6.47)$$

gdzie  $\sigma_1, \sigma_2$  są pewnymi stałymi dodatnimi. Algorytm (6.43)–(6.44) został wybrany z rodziny adaptacyjnych algorytmów Sadegha-Horowitza ze względu na to, że jest globalnie asymptotycznie stabilny. Inną ważną zaletą tego algorytmu jest zależność składnika korekcyjnego w prawie sterowania<sup>†</sup> jedynie od zadanej trajektorii przegubowej. Algorytm ten nie gwarantuje zbieżności błędu estymacji parametrów dynamiki  $\hat{\boldsymbol{\theta}}(t)$  do zera.

\*Zobacz dodatek C.

†Czyli części algorytmu sterowania wymagającej znajomości modelu dynamiki.



**Rysunek 6.9** Przebieg błędów śledzenia przy zastosowaniu adaptacyjnego algorytmu Sadegha-Horowitza z parametrami  $\mathbf{K}_d = \text{diag}\{75\}$ ,  $\mathbf{A} = \text{diag}\{2\}$ ,  $\mathbf{R} = \text{diag}\{150\}$ ,  $\mathbf{\Gamma} = \text{diag}\{20\}$  i  $K_f = 12$ .

### Przykład 6.2.5 (Manipulator EDDA)

Aby zilustrować działanie algorytmu pokażemy, podobnie jak w przypadku nieadaptacyjnym, błędy śledzenia trajektorii przegubowej (6.7) manipulatora EDDA. Wykresy błędów położenia przegubów zostały przedstawione na rysunku 6.9. ■

Z przeprowadzonych badań wynika, że spośród prezentowanych algorytmów adaptacyjnych najtrudniejszy do implementacji komputerowej jest adaptacyjny algorytm linearyzacji. Wymaga on pomiaru przyspieszeń przegubów manipulatora oraz odwracania estymowanej macierzy bezwładności. Najprostsze do praktycznego zastosowania są adaptacyjne algorytmy Bayarda-Wena i Sadegha-Horowitza, które wykorzystują do estymacji parametrów i do prawa sterowania model robota obliczany wzdłuż zadanej trajektorii.

## 6.3 Algorytmy sterowania przy strukturalnej nieznanomości modelu

Nieznanomość strukturalna modelu dynamiki manipulatora pojawia się, gdy nie jest znana postać funkcyjna zależności między zmiennymi stanu manipulatora. Przykładem takich trudnych do wyznaczenia zależności jest funkcja opisująca tarcie i opory ruchu.

W dalszych rozważaniach założymy, że dynamika manipulatora o  $n$  stopniach swobody jest opisana następującym modelem

$$\begin{aligned} \mathbf{Q}(\mathbf{q}, \boldsymbol{\theta})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\theta})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}, \boldsymbol{\theta}) + \boldsymbol{\zeta}(t) = \\ = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\theta} + \boldsymbol{\zeta}(t) = \mathbf{u}, \end{aligned} \quad (6.48)$$

gdzie  $\mathbf{q} \in \mathbb{R}^n$  oznacza wektor położenia przegubów,  $\dot{\mathbf{q}} \in \mathbb{R}^n$  jest wektorem

prędkości przegubów,  $\mathbf{Q}(\mathbf{q}, \boldsymbol{\theta})$  jest symetryczną, dodatnio określoną macierzą bezwładności,  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\theta})$  opisuje siły odśrodkowe i Coriolisa, zaś wektor  $\mathbf{D}(\mathbf{q}, \boldsymbol{\theta})$  reprezentuje oddziaływania grawitacyjne. W modelu wszystkie oddziaływania wpływające na manipulator, mające nieznaną postać, zostały potraktowane jako zaburzenia i oznaczone symbolem  $\boldsymbol{\zeta}(t)$ .

### 6.3.1 Algorytm sterowania ślizgowego

W celu przedstawienia *algorytmu sterowania ślizgowego* przyjmiemy, że dynamika manipulatora jest opisana równaniem (6.48). Załóżmy, że parametry  $\boldsymbol{\theta}$  występujące w modelu są nieznanne i dopuśćmy oddziaływanie na współrzędne przegubowe manipulatora nieznanymi zaburzeniami\*. Zaburzenia działające na współrzędną  $q_i$  mogą być opisane pewną nieznaną funkcją czasu  $\zeta_i(t)$ , gdzie  $i = 1, \dots, n$ . Zadanie sterowania będzie polegać na śledzeniu zadanej trajektorii  $\mathbf{q}_d(t)$  w przestrzeni przegubowej manipulatora, pomimo nieznanności parametrów i w obecności nieznanymi zaburzeń.

Aby możliwe było znalezienie sterowania ślizgowego zapewniającego śledzenie trajektorii, niezbędne jest posiadanie pewnej apriorycznej informacji na temat nieznanymi elementów modelu. W dalszych rozważaniach będziemy zakładali, że zarówno nieznanne parametry jak i zaburzenia są ograniczone, tzn.

$$|\theta_i| \leq \theta_i^*, \quad |\zeta_i(t)| \leq \zeta_i^*, \quad i = 1, \dots, n, \quad (6.49)$$

a wartości ograniczeń  $\theta_i^*$  i  $\zeta_i^*$  są znane. Oznacza to, że użycie algorytmu ślizgowego do sterowania manipulatora jest możliwe jedynie wtedy, gdy można przewidzieć zakres zmian nieznanymi parametrów i zaburzeń.

Algorytm sterowania ślizgowego przypomina pod względem formalnym adaptacyjny algorytm sterowania Slotine'a-Li, różni się jednak brakiem układu estymacji parametrów. Do sterowania jest wprawdzie wykorzystywana pewna ocena wektora nieznanymi parametrów  $\hat{\boldsymbol{\theta}}$ , jednak jest ona wybierana w sposób arbitralny przed rozpoczęciem procesu regulacji jako stały wektor spełniający założenie (6.49). Prawo sterowania w algorytmie ślizgowym ma postać

$$\mathbf{u} = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r) \hat{\boldsymbol{\theta}} - \mathbf{K}_d \mathbf{s} - \mathbf{K} \operatorname{sgn}(\mathbf{s}), \quad (6.50)$$

przy czym sygnały  $\dot{\mathbf{q}}_r$  i  $\mathbf{s}$  spełniają zależność (6.41). Symetryczna i dodatnio określona macierz  $\mathbf{K}_d$  ma zapewniać zbieżność algorytmu sterowania.

\* Jest to przypadek nieznanności strukturalnej modelu.

W porównaniu z algorytmem Slotine'a-Li, nowym elementem prawa sterowania (6.50) jest składnik o charakterze przełączającym  $\mathbf{K} \operatorname{sgn}(\mathbf{s}) = (k_1 \operatorname{sgn}(s_1), \dots, k_n \operatorname{sgn}(s_n))^T$ , który ma za zadanie tłumić błędy wynikłe z nieznajomości modelu. Zauważmy, że obecność składnika przełączającego w równaniu (6.50) powoduje nieciągłość względem czasu proponowanego algorytmu sterowania. Największą trudnością przy implementacji algorytmu sterowania ślizgowego jest dobór współczynników wzmocnienia  $k_i$  w taki sposób, aby algorytm stał się zbieżny.

Zastosujmy teraz sterowanie (6.50) do modelu (6.48) wyrażonego w postaci zależnej liniowo od nieznanymi parametrów

$$\begin{cases} \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\theta} + \boldsymbol{\zeta}(t) = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)\hat{\boldsymbol{\theta}} - \mathbf{K}_d\dot{\mathbf{s}} - \mathbf{K} \operatorname{sgn}(\mathbf{s}) \\ \mathbf{e} = \mathbf{q} - \mathbf{q}_d \\ \mathbf{s} = \dot{\mathbf{e}} + \boldsymbol{\Lambda}\mathbf{e} = \dot{\mathbf{q}} - \dot{\mathbf{q}}_r. \end{cases} \quad (6.51)$$

Po prostych przekształceniach i podstawieniu  $\check{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}$  uzyskujemy równanie dynamiki układu z zamkniętą pętlą sprzężenia zwrotnego

$$\mathbf{Q}(\mathbf{q}, \boldsymbol{\theta})\dot{\mathbf{s}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\theta})\mathbf{s} = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)\check{\boldsymbol{\theta}} - \mathbf{K}_d\dot{\mathbf{s}} - \mathbf{K} \operatorname{sgn}(\mathbf{s}) - \boldsymbol{\zeta}(t), \quad (6.52)$$

gdzie

$$\mathbf{q} = \mathbf{e} + \mathbf{q}_d.$$

Aby udowodnić zbieżność algorytmu sterowania ślizgowego, zastosujemy metodę funkcji Lapunowa. Rozważmy następującą funkcję

$$V(\mathbf{s}, \mathbf{e}, t) = \frac{1}{2}\mathbf{s}^T\mathbf{Q}(\mathbf{q}, \boldsymbol{\theta})\mathbf{s}, \quad (6.53)$$

zawierającą symetryczną i dodatnio określoną macierz bezwładności manipulatora  $\mathbf{Q}(\mathbf{q}, \boldsymbol{\theta})$ . Policzmy pochodną względem czasu funkcji (6.53) wzdłuż trajektorii układu (6.52), skorzystajmy z warunku skośnej symetrii (5.6) i podstawmy wyrażenie  $\mathbf{Q}(\mathbf{q}, \boldsymbol{\theta})\dot{\mathbf{s}}$  uzyskane z równania (6.52). W rezultacie, po prostych przekształceniach, otrzymamy

$$\dot{V} = -\mathbf{s}^T\mathbf{K}_d\dot{\mathbf{s}} - \mathbf{s}^T(\mathbf{K} \operatorname{sgn}(\mathbf{s}) + \boldsymbol{\zeta}(t) - \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r)\check{\boldsymbol{\theta}}). \quad (6.54)$$

Drugi składnik pochodnej (6.54) jest sumą następujących elementów

$$\begin{aligned} s_i \left( k_i \operatorname{sgn}(s_i) + \zeta_i - \sum_j Y_{ij} \ddot{\theta}_j \right) &= k_i |s_i| + \zeta_i s_i - \sum_j Y_{ij} \ddot{\theta}_j s_i = \\ &= |s_i| \left( k_i + \zeta_i \operatorname{sgn}(s_i) - \sum_j Y_{ij} \ddot{\theta}_j \operatorname{sgn}(s_i) \right). \end{aligned}$$

Zdefiniujmy teraz współczynniki  $k_i$  jako

$$k_i = \sum_j Z_{ij} \alpha_j + \zeta_i^* + \eta_i, \quad (6.55)$$

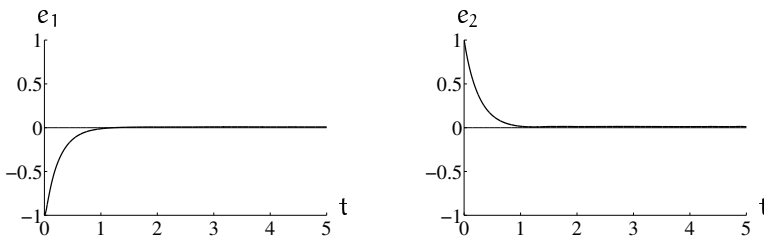
gdzie  $|Y_{ij}| \leq Z_{ij}$  wyraża ograniczenia modelu wzdłuż trajektorii odniesienia,  $\alpha_i = |\hat{\theta}_i| + \theta_i^* \geq |\ddot{\theta}_i|$  jest oszacowaniem maksymalnego błędu popełnionego podczas wyboru estymaty parametrów, zaś  $\eta_i$  jest pewną liczbą dodatnią.

Przy przyjętej definicji współczynników  $k_i$  uzyskujemy następujące oszacowanie pochodnej funkcji Lapunowa wzdłuż trajektorii układu (6.52)

$$\dot{V} \leq -\mathbf{s}^T \mathbf{K}_d \mathbf{s} - \sum_{i=1}^n \eta_i |s_i| \leq 0. \quad (6.56)$$

Z twierdzenia La Salle'a-Yoshizawy wynika, że trajektorie układu (6.52) dążą do  $\mathbf{0}$  przy  $t \rightarrow +\infty$ . Oznacza to, że błąd śledzenia  $\mathbf{e}$  spełnia równanie  $\dot{\mathbf{e}} = -\mathbf{\Lambda} \mathbf{e}$ . Poprzez wybór macierzy  $\mathbf{\Lambda}$  w taki sposób, aby jej wartości własne leżały w prawej półpłaszczyźnie zmiennej zespolonej, zagwarantowana jest asymptotyczna stabilność śledzenia. Algorytm sterowania ślizgowego wyróżnia się prostotą i niewrażliwością na zmiany parametrów manipulatora oraz zewnętrzne zakłócenia, np. efekty tarcia w przegubach, jednakże pod warunkiem, że możliwe jest oszacowanie zakresu zmian nieznanymi wielkościami. Algorytm sterowania ślizgowego może być zastosowany także wówczas, gdy oszacowanie zakłóceń zewnętrznych jest zależne od czasu, tj.  $|\zeta_i(t)| \leq \zeta_i^*(t)$ ; otrzymuje się wówczas współczynniki wzmocnienia  $k_i$  zależne w sposób jawny od czasu.

Należy jednakże pamiętać o pewnej wadze powyższego algorytmu polegającej na tym, że aby oszacować wartości współczynników  $k_i$  gwarantujące stabilność algorytmu ślizgowego, niezbędna jest znajomość maksymalnej prędkości ruchu w przegubach. W przypadku, gdy prędkość ta nie jest znana, nie można uzyskać oszacowań  $Z_{ij}$  elementów macierzy  $Y_{ij}$ . W takim wypadku wartości współczynników  $k_i$  można dobrać wyłącznie metodą symulacyjną lub eksperymentalną.



**Rysunek 6.10** Przebieg błędów śledzenia przy zastosowaniu algorytmu ślizgowego z parametrami  $\mathbf{K}_d = \text{diag}\{75\}$ ,  $\mathbf{\Lambda} = \text{diag}\{4\}$ ,  $\mathbf{K} = \text{diag}\{100\}$  i  $\hat{\boldsymbol{\theta}} = \mathbf{0}$ .

### Przykład 6.3.1 (Manipulator *EDDA*)

Dla zilustrowania działania powyższego algorytmu, pokażemy przebiegi błędów śledzenia trajektorii zadanej (6.7) dla manipulatora *EDDA*. Wykresy błędów położenia przegubów zostały przedstawione na rysunku 6.10. ■

### 6.3.2 Algorytm Qu-Dorseya — regulator PD o stałym wzmożeniu

W poprzednich podrozdziałach dokonaliśmy przeglądu układów sterowania zapewniających śledzenie zadanej trajektorii i dopuszczających kolejno coraz większy stopień nieznanymosci modelu dynamiki manipulatora. W konsekwencji, pojawia się pytanie, czy jest możliwe sprowadzenie błędu śledzenia trajektorii do zera, jeśli model manipulatora jest całkowicie nieznan (nawet wzdłuż trajektorii odniesienia). Odpowiedź na tak postawione pytanie jest negatywna. Można pokazać, że zastosowanie do modelu manipulatora (6.25) klasycznego regulatora PD\* z diagonalnymi macierzami wzmocnień  $\mathbf{K}_d > \mathbf{0}$  i  $\mathbf{K}_p > \mathbf{0}$

$$\mathbf{u} = -\mathbf{K}_d \dot{\mathbf{e}} - \mathbf{K}_p \mathbf{e} \quad (6.57)$$

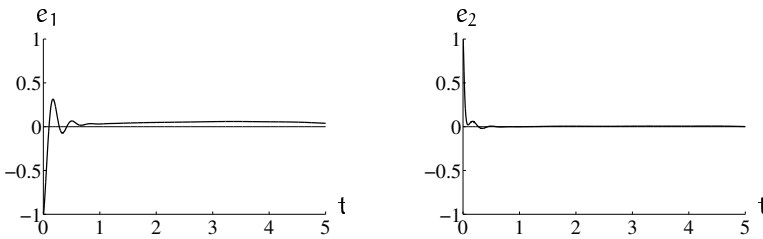
powoduje, że w układzie z zamkniętą pętlą sprzężenia zwrotnego

$$\mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e} = \mathbf{0},$$

przy  $t \rightarrow +\infty$  błąd śledzenia trajektorii

$$\mathbf{e}(t) = \mathbf{q}(t) - \mathbf{q}_d(t) \rightarrow \overline{\mathbf{B}}_{\lambda}(\mathbf{0}).$$

\*Bez generowania sterowania odniesienia  $\mathbf{u}_d$ , które wymaga chociaż częściowej znanymosci modelu dynamiki.



**Rysunek 6.11** Przebieg błędów śledzenia przy zastosowaniu algorytmu Qu-Dorsey'a ze wzmocnieniami  $\mathbf{K}_p = \text{diag}\{1500\}$  i  $\mathbf{K}_d = \text{diag}\{50\}$ .

Symbol  $\bar{\mathbf{B}}_\lambda(\mathbf{0})$  oznacza kulę domkniętą o pewnym promieniu  $\lambda > 0$  i środku w punkcie  $\mathbf{0} \in \mathbb{R}^n$ . Innymi słowy, błąd śledzenia położenia w procesie regulacji może nie zanikać do zera, ale z upływem czasu dążyć do pewnej niezerowej wartości ustalonej. Wynika stąd, że jeżeli do każdego przegubu manipulatora dołączymy lokalny regulator PD, to błąd śledzenia trajektorii będzie dążył do pewnego przedziału zawierającego 0. Wyznaczenie wartości ustalonej błędów śledzenia położenia w poszczególnych przegubach wymaga znajomości funkcji szacujących model dynamiki manipulatora. Istotną zaletą algorytmu sterowania (6.57) jest jego prostota, łatwość implementacji oraz możliwość kontrolowania obszaru, do którego dąży błąd  $\mathbf{e}(t)$ , poprzez wybór odpowiednio dużych wzmocnień  $\mathbf{K}_d$  i  $\mathbf{K}_p$ .

### Przykład 6.3.2 (Manipulator *EDDA*)

Aby zobrazować działanie algorytmu Qu-Dorsey'a, pokażemy przebiegi błędów powstałych podczas śledzenia trajektorii przez manipulator *EDDA*, opisaną wzorem (6.7). Wykresy błędów położenia przegubów zostały przedstawione na rysunku 6.11. ■

### 6.3.3 Algorytm $\lambda$ -śledzenia — regulator PD o dynamicznym wzmocnieniu

Jak pokazaliśmy, zastosowanie regulatora PD o stałym wzmocnieniu zapewnia dążenie błędu śledzenia trajektorii  $\mathbf{e}(t)$  do kuli domkniętej o promieniu  $\lambda$  i środku w punkcie  $\mathbf{0} \in \mathbb{R}^n$ . Takie działanie algorytmu sterowania będziemy nazywali  $\lambda$ -śledzeniem trajektorii\*, przy czym  $\lambda > 0$ . Obecnie proponujemy specjalny  $\lambda$ -śledzący uniwersalny algorytm sterowania

\*Śledzeniem trajektorii z błędem nie większym niż  $\lambda$ .



z tzw. martwą strefą, zapewniający następującą własność

$$P_1 \mathbf{e}(t) + P_2 \dot{\mathbf{e}}(t) \rightarrow \bar{\mathbf{B}}_\lambda(\mathbf{0}), \quad 0 < P_1, P_2 \in \mathbb{R}, \quad \text{gdy } t \rightarrow +\infty,$$

gdzie  $\mathbf{e}(t) = \mathbf{q}(t) - \mathbf{q}_d(t)$ ,  $\dot{\mathbf{e}}(t) = \dot{\mathbf{q}}(t) - \dot{\mathbf{q}}_d(t)$ . Łatwo pokazać, że zbieżność kombinacji liniowej błędów  $\mathbf{e}$  i  $\dot{\mathbf{e}}$  do kuli o promieniu  $\lambda$  i środka w zerze implikuje zbieżność błędu  $\mathbf{e}(t)$  do kuli o promieniu  $\lambda/P_1$ .

W celu wyprowadzenia algorytmu  $\lambda$ -śledzenia trajektorii rozważymy model dynamiki manipulatora postaci (6.25). Załóżmy, że  $\lambda > 0$  i podajmy na wejście  $\mathbf{u}$  manipulatora opisanego równaniem (6.25) sterowanie

$$\mathbf{u}(t) = -k(t)\mathbf{E}(t), \quad (6.58)$$

gdzie

$$\mathbf{E}(t) = P_1 \mathbf{e}(t) + P_2 \dot{\mathbf{e}}(t). \quad (6.59)$$

Wzmocnienie  $k(t)$  sterowania będziemy adaptować zgodnie z regułą

$$\dot{k} = d_\lambda(\mathbf{E})\|\mathbf{E}\| = \begin{cases} (\|\mathbf{E}\| - \lambda)\|\mathbf{E}\|, & \text{jeżeli } \|\mathbf{E}\| > \lambda, \\ 0, & \text{jeżeli } \|\mathbf{E}\| \leq \lambda, \end{cases} \quad (6.60)$$

przy założeniu, że  $k(0) > 0$ . Układ z zamkniętą pętlą sprzężenia zwrotnego jest opisany następującymi równaniami

$$\begin{cases} \mathbf{Q}(\mathbf{q})\ddot{\mathbf{e}} = -\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{e}} - kP_1 \mathbf{e}(t) - kP_2 \dot{\mathbf{e}}(t) - \mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}}_d - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}_d - \mathbf{D}(\mathbf{q}) \\ \dot{k} = d_\lambda(\mathbf{E})\|\mathbf{E}\| \\ \mathbf{q} = \mathbf{q}_d + \mathbf{e}. \end{cases} \quad (6.61)$$

W dalszych rozważaniach pokażemy, że układ (6.61) posiada następujące własności:

$$\begin{aligned} \mathbf{E}(t) = P_1 \mathbf{e}(t) + P_2 \dot{\mathbf{e}}(t) &\rightarrow \bar{\mathbf{B}}_\lambda(\mathbf{0}) \quad \text{przy } t \rightarrow +\infty, \\ \lim_{t \rightarrow +\infty} k(t) &< \infty. \end{aligned} \quad (6.62)$$

W tym celu rozpatrzmy funkcję Lapunowa postaci

$$V(\mathbf{E}, k) = V_\lambda(\mathbf{E}) + \frac{a}{2} \left( k - \frac{b+1}{a} \right)^2 = \frac{1}{2} d_\lambda^2(\mathbf{E}) + \frac{a}{2} \left( k - \frac{b+1}{a} \right)^2, \quad (6.63)$$

przy czym

$$V_\lambda(\mathbf{E}) = \frac{1}{2} d_\lambda^2(\mathbf{E}) = \begin{cases} \frac{1}{2}(\|\mathbf{E}\| - \lambda)^2, & \text{jeżeli } \|\mathbf{E}\| > \lambda, \\ 0, & \text{jeżeli } \|\mathbf{E}\| \leq \lambda, \end{cases} \quad (6.64)$$

zaś  $a$  i  $b$  są pewnymi znanymi liczbami dodatnimi. Dowód zbieżności algorytmu  $\lambda$ -śledzenia wymaga przyjęcia założenia o ograniczonej prędkości manipulatora. Przy takim założeniu można pokazać, że prawdziwa jest następująca nierówność

$$\dot{V} \leq -\dot{k} \leq 0. \quad (6.65)$$

Z (6.65), (6.63) i (6.64) wynika, że trajektorie  $\mathbf{e}(t)$ ,  $k(t)$  są ograniczone. Co więcej, z twierdzenia La Salle'a-Yoshizawy wnioskujemy, że trajektorie układu z zamkniętą pętlą sprzężenia zwrotnego dążą do zbioru

$$\|\mathbf{E}\| \leq \lambda, \quad (6.66)$$

co jest równoważne własności (6.62). Po prostych przekształceniach można pokazać, że jeśli zachodzi (6.66), to dla każdego przegubu mamy

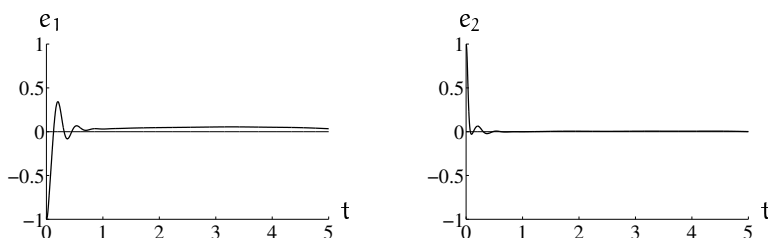
$$e_i(t) \rightarrow \left[ -\frac{\lambda}{P_1}, \frac{\lambda}{P_1} \right] \quad \text{przy } t \rightarrow +\infty,$$

czyli każda składowa błędu  $\mathbf{e}(t)$  dąży do znanego przedziału domkniętego.

Algorytm  $\lambda$ -śledzenia jest odporny na zakłócenia. Występujący w nim parametr  $\lambda > 0$ , czyli szerokość martwej strefy (strefy nieczułości), ustalamy przed przystąpieniem do regulacji na podstawie przewidywanej wielkości zakłóceń. Jeśli tylko wartość wzmocnienia  $P_1$  będzie dostatecznie duża, można zagwarantować zbieżność błędu  $e_i(t)$  do małego, wybranego przez nas otoczenia punktu 0, dla każdego  $i = 1, \dots, n$ . Należy podkreślić, że szerokość przedziału, w którym znajduje się błąd śledzenia położenia, zależy jedynie od doboru wartości parametrów  $\lambda$  i  $P_1$  algorytmu sterowania, a nie zależy od stopnia znajomości dynamiki manipulatora.

### Przykład 6.3.3 (Manipulator *EDDA*)

Działanie algorytmu  $\lambda$ -śledzenia zilustrujemy na przykładzie zadania śledzenia trajektorii opisanej wzorem (6.7) dla manipulatora *EDDA*. Wykresy błędów położenia przegubów zostały przedstawione na rysunku 6.12. W czasie trwania symulacji wzmocnienie układu uniwersalnego ustaliło się na wartości  $k_{ust} = 57$ . ■



**Rysunek 6.12** Przebieg błędów śledzenia przy zastosowaniu algorytmu  $\lambda$ -śledzenia z parametrami  $P_1 = 30$ ,  $P_2 = 1$  i  $\lambda = 0.5$ .

Spośród algorytmów sterowania przedstawionych w tym rozdziale, najprostszym i najczęściej wykorzystywanym w praktyce jest algorytm Qu-Dorsey'a, a więc statyczny regulator PD. Algorytm  $\lambda$ -śledzenia jest nieco bardziej skomplikowany, bowiem wymaga obliczania wzmocnienia dynamicznego, ale za to daje możliwość kontrolowania dokładności śledzenia. Algorytm ślizgowy jest najtrudniejszy do praktycznej implementacji, poza tym prowadzi często do wzbudzenia w układzie sterowania drgań granicznych, polegających na przełączaniu sterowania z dużą częstotliwością w pobliżu powierzchni ślizgu.

## 6.4 Komentarze i uwagi bibliograficzne

Układem dysypatywnym nazywa się układ, w którym przyrost energii wewnętrznej jest mniejszy od energii dostarczonej do układu. Znaczenie własności pasywności i dysypatywności dla badania stabilności układów, w tym układów sterowania robotów, zostało objaśnione w pracach [SL88, Spo98]. Koncepcja adaptacyjnych układów sterowania nie wykorzystujących identyfikacji parametrów została sformułowana przez Marelsa [Mar84], Mårtenssona [Mår85] oraz Morse'a [Mor83]. W pracy [Mor83] Morse postawił hipotezę, że znajomość znaku wyrażenia  $\mathbf{c}^T \mathbf{b}^*$  jest warunkiem koniecznym zbieżności algorytmów adaptacyjnej stabilizacji. Hipotezę Morse'a sfalsyfikował Nussbaum [Nus83], który wprowadził pojęcie funkcji przełączającej i zdefiniował adaptacyjny układ stabilizacji dla systemów liniowych pierwszego rzędu, jednowejściowych i jednowyjściowych, przy czym znak  $\mathbf{c}^T \mathbf{b}$  był nieznanym. Pomysł ten rozwinęli Byrnes i Willems [BW84]. W przypadku układów (6.2) o wielu wejściach i wielu wyjściach (wówczas  $m = p > 1$ ,

\*Zobacz wzór (6.2).

a wektory  $\mathbf{b}$ ,  $\mathbf{c}$  zastępujemy macierzami  $\mathbf{B}$ ,  $\mathbf{C}$ ), Ilchmann i Townley [IT93] oraz Schmid [Sch91] pokazali, że przy założeniu, że widmo macierzy  $\mathbf{CB}$  leży w dodatniej półpłaszczyźnie płaszczyzny zespolonej, układ sterowania

$$\mathbf{u}(t) = -\mathbf{k}(t)\mathbf{y}(t), \quad \dot{\mathbf{k}}(t) = \|\mathbf{y}(t)\|^2$$

jest uniwersalnym adaptacyjnym układem stabilizacji. Najogólniejsze rozwiązanie problemu syntezy uniwersalnego układu sterowania, nie wymagające założeń na temat minimalnofazowości ani stopnia względnego obiektu podano w pracach [Mär85, MP96]. Od początku lat dziewięćdziesiątych nasiliły się badania mające na celu rozszerzenie klasy obiektów, dla których uniwersalne układy sterowania działają poprawnie. Ilchmann [Ilc93, IT93] rozszerzył klasę obiektów stabilizowalnych przy pomocy uniwersalnych układów sterowania o układy o wielu wejściach i wyjściach, a także o układy o stopniu względnym 2 oraz o układy z nieliniowym zaburzeniem. Praetzel-Wolters [PW93] oraz Schmid [Sch91] zdefiniowali uniwersalne układy sterowania dla obiektów liniowych, niestacjonarnych, oddziałujących na siebie poprzez interakcje. Dalsze prace nad uniwersalnymi adaptacyjnymi układami sterowania zmierzają w kierunku rozszerzenia klasy obiektów sterowania o układy nieliniowe, jak również układy o wyższych stopniach względnych i układy z wektorowym wzmocnieniem w pętli sprzężenia zwrotnego. Udaną próbą zastosowania uniwersalnych układów sterowania do sterowania obiektami nieliniowymi było użycie tych układów do sterowania manipulatorów [Maz96a, Maz96b], robotów mobilnych [MH97, Maz98] i procesów chemicznych [AI95]. Algorytm linearyzacji statycznej w wersji adaptacyjnej i nieadaptacyjnej można znaleźć w pracy [Ber93]. Dynamika silników napędzających przeguby została uwzględniona w artykule [T<sup>+</sup>91]. Klasa nieadaptacyjnych algorytmów Wena-Bayarda została przedstawiona w artykule [WB88], zaś ich wersje adaptacyjne przedstawiono w [BW88]. Dowody stabilności wszystkich algorytmów podanych przez Wena i Bayarda oraz uniwersalnego adaptacyjnego algorytmu sterowania opierają się na lemacie Wena-Bayarda (patrz dodatek B), który jest w rzeczywistości twierdzeniem o lokalnej stabilności. Pojęcie stabilności półglobalnej zostało wprowadzone w [TP94]. Wśród adaptacyjnych algorytmów obliczanego momentu należy wymienić algorytm Middletona-Goodwina [MG88], w którym poprzez zastosowanie odpowiedniego filtra udało się uniknąć pomiaru przyspieszenia manipulatora, oraz algorytm Sponga-Ortegi [SO90] zapewniający odwracalność estymowanej macierzy bezwładności manipulatora. Spośród algorytmów wykorzystujących pojęcie zmiennej śli-

zgu, algorytm Slotine'a-Li w wersji nieadaptacyjnej i adaptacyjnej oraz algorytm ślizgowy można znaleźć w [SL87, SL88]. Algorytm sterowania ślizgowego robota IRb-6 zaproponowali Gosiewski i Szykiewicz w referacie [GS88]. W pracy [SL89] wprowadzono zmodyfikowane algorytmy estymacji parametrów, które zapewniają zbieżność estymacji, o ile jest spełniony warunek trwałego wzbudzenia. Z kolei, klasa algorytmów sterowania wprowadzona przez Sadegha-Horowitza, w tym oba algorytmy przedstawione w tym rozdziale, znajduje się w pracy [SH90]. Algorytm Qu-Dorsey'a wraz z dowodem został przedstawiony w [QD91]. Przykładem zastosowania układów uniwersalnych do sterowania manipulatora jest algorytm  $\lambda$ -śledzenia trajektorii [Maz96b]. Dowód zbieżności tego algorytmu bazuje na twierdzeniu La Salle'a-Yoshizawy, które zostało przedstawione w dodatku B. Możliwość liniowej parametryzacji modelu dynamiki manipulatora ma znaczenie fundamentalne [SV97, Koz98]. W celu przeanalizowania działania algorytmów sterowania opisanych w tym rozdziale wykonaliśmy badania eksperymentalne na manipulatorze *EDDA*. Badania pokazały, że żaden z algorytmów nie gwarantował redukcji do zera błędu śledzenia, a jedynie zapewniał ustalenie się błędu śledzenia na pewnej nieznaczącej wartości. Wynikało to prawdopodobnie z nieliniowości układów napędowych manipulatora *EDDA*, odgrywającej rolę nieliniowego zaburzenia przyjętego modelu dynamiki. Algorytmy uniwersalne bez strefy nieczułości okazały się najbardziej wrażliwe na obecność tego zaburzenia, co objawiło się utratą przez nie stabilności. Jedynymi algorytmami, które umożliwiły istotne ograniczenie ustalonej wartości błędu śledzenia były: algorytm Qu-Dorsey'a i algorytm  $\lambda$ -śledzenia, przy czym ten ostatni algorytm w przeprowadzonych eksperymentach zapewniał największą dokładność śledzenia.

## Literatura

- [AI95] F. Allgoewer i A. Ilchmann, Multivariable adaptive  $\lambda$ -tracking for nonlinear chemical processes. W: *Proc. Europ. Contr. Conf.*, strony 1645–1651, Rome, 1995.
- [Ber93] H. Berghuis, *Model-Based Robot Control: from Theory to Practice*. Rozprawa doktorska, University Twente, Enschede, 1993.
- [BW84] C. I. Byrnes i J. C. Willems, Adaptive stabilization of multivariable linear systems. W: *Proc. IEEE Conf. Decision Control*, vol. 2, strony 1574–1577, 1984.

- [BW88] D. S. Bayard i J. T. Wen, New class of control laws for robotic manipulators: Part 2 — adaptive case. *Int. J. Control*, **47**(5):1387–1406, 1988.
- [GS88] A. Gosiewski i W. Szykiewicz, Zastosowanie algorytmu ruchu ślizgowego do układu sterowania manipulatora robota. W: *Materiały 2 Krajowej Konferencji Robotyki*, vol. **1**, strony 141–150, Wrocław, 1988.
- [Ilc93] A. Ilchmann, *Non-Identifier-Based High-Gain Adaptive Control*. Springer-Verlag, London, 1993.
- [IT93] A. Ilchmann i S. Townley, Simple adaptive stabilization of high-gain stabilizable systems. *Systems & Contr. Lett.*, **20**:189–198, 1993.
- [Koz98] K. Kozłowski, *Modelling and Identification in Robotics*. Springer-Verlag, Berlin, 1998.
- [Mar84] I. Mareels, A simple self-tuning controller for stably invertible systems. *Systems & Contr. Lett.*, 4:5–16, 1984.
- [Már85] B. Mårtensson, The order of any stabilizing regulator is sufficient a priori information for adaptive stabilization. *Systems & Contr. Lett.*, **6**:87–91, 1985.
- [Maz96a] A. Mazur, *Algorytmy sterowania robotów oparte na zasadzie uniwersalnego adaptacyjnego układu sterowania*. Rozprawa doktorska, Instytut Cybernetyki Technicznej, Politechnika Wrocławska, 1996.
- [Maz96b] A. Mazur, Uniwersalny adaptacyjny  $\lambda$ -śledzący układ sterowania robota. W: *Materiały 5 Krajowej Konferencji Robotyki*, strony 124–131, Świeradów Zdrój, 1996.
- [Maz98] A. Mazur, Universal adaptive tracking controller for nonholonomic multibody wheeled mobile robots. W: *Proc. MMAR Symposium*, vol. **3**, strony 937–944, Międzyzdroje, 1998.
- [MG88] R. H. Middleton i G. C. Goodwin, Adaptive computed torque control for rigid link manipulations. *Systems & Contr. Lett.*, **10**:9–16, 1988.
- [MH97] A. Mazur i R. Hossa, Universal adaptive  $\lambda$ -tracking controller for wheeled mobile robots. W: *Proc. IFAC SyRoCo Conference*, vol. **1**, strony 33–37, Nantes, 1997.
- [Mor83] A. S. Morse, Recent problems in parameter adaptive control. W: I. D. Landau, (red.), *Outils et Modèles Mathématiques pour l'Automatique, l'Analyse de Systèmes et le Traitement du Signal*, strony 733–740. Editions du CNRS3, Paris, 1983.
- [MP96] I. Mareels i J. W. Polderman, *Adaptive Systems: An Introduction*. Birkhäuser, Boston, 1996.

- [Nus83] R. D. Nussbaum, Some remarks on a conjecture in parameter adaptive control. *Systems & Contr. Lett.*, **3**:243–246, 1983.
- [PW93] D. Praetzel-Wolters, Adaptive control without parameter identification. W: *Proc. Int. Symp. MTNS*, strony 311–337, 1993.
- [QD91] Z. Qu i J. Dorsey, Robust tracking control of robots by a linear feedback law. *IEEE Trans. Robotics Automat.*, **36**(9):1081–1084, 1991.
- [Sch91] S. Schmid, *Adaptive Synchronization of Interconnected Systems*. Rozprawa doktorska, University of Kaiserslautern, 1991.
- [SH90] N. Sadegh i R. Horowitz, Stability and robustness analysis of a class of adaptive controllers for robotic manipulators. *Int. J. Robotics Research*, **9**(3):74–94, 1990.
- [SL87] J. J. E. Slotine i W. Li, On the adaptive control of robot manipulators. *Int. J. Robotics Research*, **6**(3):49–59, 1987.
- [SL88] J. J. E. Slotine i W. Li, Adaptive manipulator control: A case study. *IEEE Trans. Autom. Contr.*, **33**(11):995–1003, 1988.
- [SL89] J. J. E. Slotine i W. Li, Composite adaptive control of robot manipulators. *Automatica*, **25**(4):509–529, 1989.
- [SO90] M. Spong i R. Ortega, On adaptive inverse dynamics control of rigid robots. *IEEE Trans. Autom. Contr.*, **35**(1):92–95, 1990.
- [Spo98] M. W. Spong, On feedback linearization of robot manipulators and riemannian curvature. W: J. Bailleul, S. S. Sastry i H. J. Sussmann, (red.), *Essays on Mathematical Robotics*, strony 185–202. Springer-Verlag, New York, 1998.
- [SV97] M. Spong i M. Vidyasagar, *Dynamika i sterowanie robotów*. WNT, Warszawa, 1997.
- [T<sup>+</sup>91] T. J. Tarn et al., Effect of motor dynamics on nonlinear feedback robot arm control. *IEEE Trans. Robotics Automat.*, **7**(1):114–122, 1991.
- [TP94] A. Teel i L. Praly, Global stabilizability and observability imply semi-global stabilizability by output-feedback. *Systems & Contr. Lett.*, **22**(5):313–325, 1994.
- [WB88] J. T. Wen i D. S. Bayard, New class of control laws for robotic manipulators: Part 1 — non-adaptive case. *Int. J. Control*, **47**(5):1361–1386, 1988.





## Rozdział 7

# Algorytmy sterowania manipulatorów o elastycznych przegubach

W manipulatorze mogą wystąpić dwa rodzaje sił elastyczności powodujących powstanie niedokładności śledzenia trajektorii: elastyczność ramion i elastyczność przegubów. Właściwe rozpoznanie typu elastyczności i jej uwzględnienie w modelu dynamiki umożliwi istotną poprawę jakości sterowania.

Elastyczność ramion pojawia się w przypadku manipulatorów o ramionach zbudowanych z lekkich materiałów oraz wtedy, gdy ramiona manipulatora są znacznej długości. Taka konstrukcja manipulatora powoduje uginanie się ramion podczas pracy i powstanie niedokładności pozycjonowania. Uwzględnienie elastyczności ramion prowadzi do skomplikowanych modeli dynamiki opisanych równaniami różniczkowymi cząstkowymi. Oprócz elastyczności ramion może wystąpić elastyczność przegubów manipulatora, zwłaszcza w przypadku manipulatorów o napędzie niebezpośrednim. Skutkiem elastyczności przegubów jest pojawienie się różnicy między położeniem układu napędowego (wał silnika) a położeniem napędzanego przegubu. Za błędy powstające przy przekazywaniu napędu odpowiadają takie zjawiska fizyczne jak tarcie, poślizg i luzy w przekładniach, efekty histerezy oraz siły sprężystości. Model manipulatora o elastycznych przegubach jest wprawdzie skończenie wymiarowy\*, ale znacznie bardziej złożony od modelu manipulatora sztywnego. W dalszych rozważaniach ograniczymy się do sytuacji,

---

\*W przeciwieństwie do modelu manipulatora z elastycznymi ramionami.

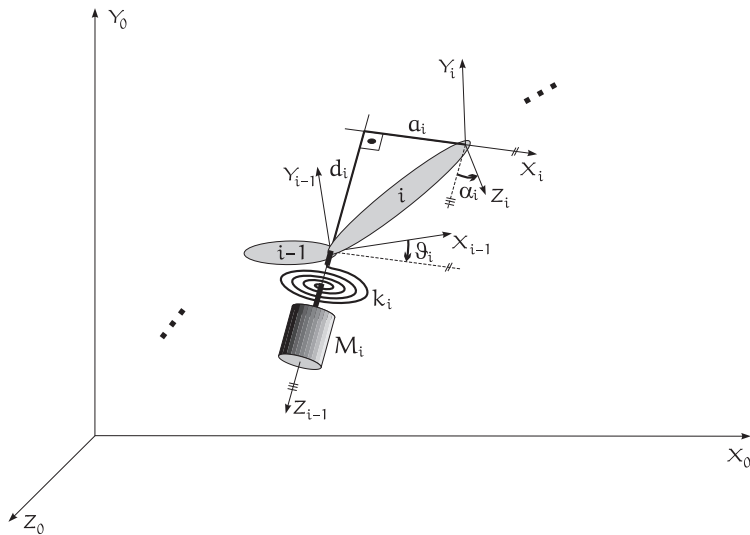
w których przyczyną elastyczności przegubów są wyłącznie siły sprężystości. Przy ograniczeniu uwagi do małych, sprężystych deformacji przegubów, uzasadnione jest założenie, że siły sprężystości zależą od tych deformacji w sposób liniowy.

W porównaniu ze sztywnym manipulatorem, model manipulatora o elastycznych przegubach wymaga do kompletnego opisu stanu manipulatora dwukrotnie większej liczby współrzędnych uogólnionych. W modelu manipulatora z elastycznymi przegubami liczba wejść sterujących jest mniejsza niż liczba mechanicznych stopni swobody. Z tych powodów, zadanie sterowania staje się znacznie trudniejsze niż zadanie sterowania dla manipulatorów sztywnych. W szczególności, implementacja algorytmu sterowania opartego na statycznym sprzężeniu zwrotnym wymaga instalacji sensorów mierzących zarówno położenie wałów silników, jak i położenie przegubów.

Niniejszy rozdział przedstawia wybrane algorytmy sterowania zapewniające śledzenie trajektorii przegubowej manipulatorów o sztywnych ramionach i elastycznych przegubach. Spośród algorytmów sterowania manipulatorów o elastycznych przegubach omówimy wyłącznie algorytmy nieadaptacyjne, a ich działanie zilustrujemy na przykładach.

## 7.1 Dynamika manipulatora o elastycznych przegubach

Podobnie jak w przypadku manipulatora sztywnego, równania dynamiki manipulatora o elastycznych przegubach uzyskamy przy pomocy formalizmu Lagrange'a przedstawionego w podrozdziale 2.2. Załóżmy, że manipulator posiada  $n$  stopni swobody i że z każdym stopniem swobody jest związany układ napędowy. W takiej sytuacji, do opisu dynamiki manipulatora będą potrzebne współrzędne uogólnione  $\mathbf{q}^1 = (q_1^1, \dots, q_n^1)^T \in \mathbb{R}^n$  określające położenia przegubów, oraz  $\mathbf{q}^2 = (q_1^2, \dots, q_n^2)^T \in \mathbb{R}^n$ , które definiują położenia wałów silników napędzających. Odpowiednie prędkości uogólnione oznaczymy symbolami  $\dot{\mathbf{q}}^1, \dot{\mathbf{q}}^2 \in \mathbb{R}^n$ . Elastyczność transmisji napędu od układu napędowego do przegubu będziemy modelować przy pomocy liniowej sprężyny poddanej siłom skręcającym. Przyjmijmy, że układem napędowym ramienia nr  $i$  manipulatora jest silnik elektryczny, którego stojan stanowi część  $(i-1)$ -szego ramienia, a wirnik obraca się razem z  $i$ -tym ogniwem wokół osi  $Z_{i-1}$  układu współrzędnych  $X_{i-1}Y_{i-1}Z_{i-1}$  (zobacz rysunek 7.1).



**Rysunek 7.1** Schemat połączenia elastycznego pomiędzy dwoma ramionami manipulatora o elastycznych przegubach.

Rozważania analogiczne do tych, które przeprowadziliśmy w podrozdziale 5.2, pozwalają uzyskać następujące wyrażenia na energię kinetyczną  $i$ -tego ramienia i  $i$ -tego układu napędowego

$$\begin{aligned} K_{L_i}(\mathbf{q}^{1i}, \dot{\mathbf{q}}^{1i}) &= \frac{1}{2} \text{tr} \left( \dot{\mathbf{A}}_0^i(\mathbf{q}^{1i}) \mathbf{J}_{L_i} \dot{\mathbf{A}}_0^{iT}(\mathbf{q}^{1i}) \right) = \\ &= \frac{1}{2} \sum_{j,k=1}^n \text{tr} \left( \frac{\partial \mathbf{A}_0^i(\mathbf{q}^{1i})}{\partial q_j^1} \mathbf{J}_{L_i} \left( \frac{\partial \mathbf{A}_0^i(\mathbf{q}^{1i})}{\partial q_k^1} \right)^T \right) \dot{q}_j^1 \dot{q}_k^1, \end{aligned} \quad (7.1)$$

$$\begin{aligned} K_{M_i}(\mathbf{q}^{1i-1}, q_i^2, \dot{\mathbf{q}}^{1i-2}, \dot{q}_i^2) &= \frac{1}{2} \text{tr} \left( \dot{\mathbf{A}}_0^{i-1}(\mathbf{q}^{1i-1}) \hat{\mathbf{J}}_{M_i} \left( \dot{\mathbf{A}}_0^{i-1} \right)^T(\mathbf{q}^{1i-1}) \right) + \\ &+ \text{tr} \left( \dot{\mathbf{A}}_0^{i-1}(\mathbf{q}^{1i-1}) \hat{\mathbf{J}}_{M_i} \begin{bmatrix} -[\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} \left( \mathbf{A}_0^{i-1} \right)^T(\mathbf{q}^{1i-1}) \right) \dot{q}_i^2 + \frac{1}{2} I_i (\dot{q}_i^2)^2. \end{aligned} \quad (7.2)$$

W powyższych wzorach  $\mathbf{J}_{L_i}$  oznacza macierz inercji  $i$ -tego ramienia,  $\mathbf{J}_{M_i}$ ,  $I_i$  są, odpowiednio, macierzą inercji i momentem bezwładności  $i$ -tego wirnika względem osi  $Z_{i-1}$  układu  $X_{i-1} Y_{i-1} Z_{i-1}$ ,  $\hat{\mathbf{J}}_{M_i} = \mathbf{Rot}(Z, q_i^2) \mathbf{J}_{M_i} \mathbf{Rot}^T(Z, q_i^2)$ , natomiast  $\mathbf{A}_0^i$  oznacza macierz transformacji układu podstawowego  $X_0 Y_0 Z_0$  w układ  $X_i Y_i Z_i$ ,  $\mathbf{q}^{1i} = (q_1^1, \dots, q_i^1)^T$ .

Oznaczmy pierwszy i drugi składnik prawej strony wyrażenia (7.2) jako  $K_{M_i1}$  i  $K_{M_i2}$ . Przy założeniu, że wirnik  $i$ -tego silnika ma postać walca\*

\*Ogólniej — wykazuje symetrię obrotową względem osi  $Z$ .

obrcającego się wokół osi  $Z_{i-1}$  układu  $X_{i-1}Y_{i-1}Z_{i-1}$ , którego środek masy pokrywa się z początkiem tego układu, nietrudno pokazać, że macierz inercji  $\mathbf{J}_{M_i}$  jest diagonalna i ma postać

$$\mathbf{J}_{M_i} = \text{diag}\{\alpha_i, \alpha_i, \beta_i, \gamma_i\}.$$

Dla macierzy  $\mathbf{J}_{M_i}$  zachodzi związek  $\hat{\mathbf{J}}_{M_i} = \mathbf{J}_{M_i}$ , a zatem

$$\begin{aligned} K_{M_i1} &= \frac{1}{2} \text{tr} \left( \dot{\mathbf{A}}_0^{i-1}(\mathbf{q}^{1i-1}) \mathbf{J}_{M_i} \left( \dot{\mathbf{A}}_0^{i-1} \right)^T (\mathbf{q}^{1i-1}) \right) = \\ &= \frac{1}{2} \sum_{j,k=1}^n \text{tr} \left( \frac{\partial \mathbf{A}_0^{i-1}(\mathbf{q}^{1i-1})}{\partial q_j^1} \mathbf{J}_{M_i} \left( \frac{\partial \mathbf{A}_0^{i-1}(\mathbf{q}^{1i-1})}{\partial q_k^1} \right)^T \right) \dot{q}_j^1 \dot{q}_k^1. \end{aligned} \quad (7.3)$$

Podobnie

$$\begin{aligned} K_{M_i2} &= \text{tr} \left( \dot{\mathbf{A}}_0^{i-1}(\mathbf{q}^{1i-1}) \mathbf{J}_{M_i} \begin{bmatrix} -[\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} (\mathbf{A}_0^{i-1})^T (\mathbf{q}^{1i-1}) \right) \dot{q}_i^2 \\ &= \sum_{j=1}^n \text{tr} \left( \frac{\partial \mathbf{A}_0^{i-1}(\mathbf{q}^{1i-1})}{\partial q_j^1} \begin{bmatrix} -\alpha_i [\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} (\mathbf{A}_0^{i-1})^T (\mathbf{q}^{1i-1}) \right) \dot{q}_j^1 \dot{q}_i^1. \end{aligned} \quad (7.4)$$

Całkowita energia kinetyczna manipulatora z układami napędowymi jest sumą energii (7.1) i (7.2) i ma postać formy kwadratowej

$$\mathbf{K}(\mathbf{q}^1, \mathbf{q}^2, \dot{\mathbf{q}}^1, \dot{\mathbf{q}}^2) = \frac{1}{2} \begin{pmatrix} \dot{\mathbf{q}}^1 \\ \dot{\mathbf{q}}^2 \end{pmatrix}^T \begin{bmatrix} \mathbf{Q}(\mathbf{q}^1) & \mathbf{S}(\mathbf{q}^{1n-1}) \\ \mathbf{S}^T(\mathbf{q}^{1n-1}) & \mathbf{I} \end{bmatrix} \begin{pmatrix} \dot{\mathbf{q}}^1 \\ \dot{\mathbf{q}}^2 \end{pmatrix}. \quad (7.5)$$

Elementy macierzy  $\mathbf{Q}(\mathbf{q}^1)$  są zdefiniowane przez wzory (7.1), (7.3) w następujący sposób (porównaj z (5.13))

$$\begin{aligned} Q_{jk}(\mathbf{q}^1) &= \sum_{i=1}^{n-1} \text{tr} \left( \frac{\partial \mathbf{A}_0^i(\mathbf{q}^{1i})}{\partial q_j^1} (\mathbf{J}_{L_i} + \mathbf{J}_{M_{i+1}}) \left( \frac{\partial \mathbf{A}_0^i(\mathbf{q}^{1i})}{\partial q_k^1} \right)^T \right) + \\ &+ \text{tr} \left( \frac{\partial \mathbf{A}_0^n(\mathbf{q}^1)}{\partial q_j^1} \mathbf{J}_{L_n} \left( \frac{\partial \mathbf{A}_0^n(\mathbf{q}^1)}{\partial q_k^1} \right)^T \right). \end{aligned} \quad (7.6)$$

Do obliczenia elementów macierzy  $\mathbf{S}(\mathbf{q}^{1n-1})$  wykorzystamy wyrażenie (7.4), z którego otrzymujemy

$$S_{jk}(\mathbf{q}^{1n-1}) = \text{tr} \left( \frac{\partial \mathbf{A}_0^{k-1}(\mathbf{q}^{1k-1})}{\partial q_j^1} \begin{bmatrix} -\alpha_i [\mathbf{e}_3] & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix} (\mathbf{A}_0^{k-1})^T (\mathbf{q}^{1k-1}) \right). \quad (7.7)$$

Co więcej, ponieważ

$$\frac{\partial \mathbf{A}_0^{k-1}(\mathbf{q}^{1k-1})}{\partial q_j^1} = \mathbf{0} \quad \text{dla } j = k, k+1, \dots, n,$$

macierz  $\mathbf{S}(\mathbf{q}^{1n-1})$  jest górnio-trójkątna, tzn.

$$\mathbf{S}(\mathbf{q}^{1n-1}) = \begin{bmatrix} 0 & S_{12}(q_1^1) & S_{13}(q_1^1, q_2^1) & \cdots & S_{1n}(q_1^1, \dots, q_{n-1}^1) \\ 0 & 0 & S_{23}(q_2^1) & \cdots & S_{2n}(q_2^1, \dots, q_{n-1}^1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & S_{n-1n}(q_{n-1}^1) \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}. \quad (7.8)$$

Macierz  $\mathbf{I}$  występująca w formie kwadratowej (7.5) składa się z momentów bezwładności wirników,

$$\mathbf{I} = \text{diag}\{I_1, I_2, \dots, I_n\}.$$

Energia potencjalna manipulatora o elastycznych przegubach składa się z energii potencjalnej ramion wraz z układami napędowymi oraz z energii potencjalnej elastyczności przegubów. Zakładając, że energia elastyczności

$$V_E(\mathbf{q}^1, \mathbf{q}^2) = \frac{1}{2}(\mathbf{q}^1 - \mathbf{q}^2)^\top \mathbf{K}(\mathbf{q}^1 - \mathbf{q}^2), \quad (7.9)$$

gdzie  $\mathbf{K} = \text{diag}\{k_1, \dots, k_n\}$  jest macierzą współczynników elastyczności przegubów, i korzystając z zależności (5.15), zapiszemy energię potencjalną manipulatora w postaci

$$V(\mathbf{q}^1, \mathbf{q}^2) = - \sum_{i=1}^n m_i \mathbf{g}^\top \mathbf{A}_0^i(\mathbf{q}^{1i}) \mathbf{R}_i + V_E(\mathbf{q}^1, \mathbf{q}^2). \quad (7.10)$$

Z równań Eulera-Lagrange'a dla lagranżianu

$$L(\mathbf{q}^1, \mathbf{q}^2, \dot{\mathbf{q}}^1, \dot{\mathbf{q}}^2) = K(\mathbf{q}^1, \mathbf{q}^2, \dot{\mathbf{q}}^1, \dot{\mathbf{q}}^2) - V(\mathbf{q}^1, \mathbf{q}^2),$$

gdzie  $K$  jest dane przez (7.5), a  $V$  przez (7.10) otrzymujemy równania dynamiki manipulatora o elastycznych przegubach

$$\begin{cases} \mathbf{Q}(\mathbf{q}^1) \ddot{\mathbf{q}}^1 + \mathbf{S}(\mathbf{q}^{1n-1}) \ddot{\mathbf{q}}^2 + \dot{\mathbf{S}}(\mathbf{q}^{1n-1}) \dot{\mathbf{q}}^2 - \frac{1}{2} \frac{\partial}{\partial \mathbf{q}^1} (\dot{\mathbf{q}}^1 \mathbf{S}(\mathbf{q}^{1n-1}) \dot{\mathbf{q}}^2) + \\ \quad + \mathbf{C}(\mathbf{q}^1, \dot{\mathbf{q}}^1) \dot{\mathbf{q}}^1 + \mathbf{D}(\mathbf{q}^1) + \mathbf{K}(\mathbf{q}^1 - \mathbf{q}^2) = \mathbf{0} \\ \mathbf{I} \ddot{\mathbf{q}}^2 + \mathbf{S}^\top(\mathbf{q}^{1n-1}) \ddot{\mathbf{q}}^1 + \dot{\mathbf{S}}^\top(\mathbf{q}^{1n-1}) \dot{\mathbf{q}}^1 - \mathbf{K}(\mathbf{q}^1 - \mathbf{q}^2) = \mathbf{u}. \end{cases} \quad (7.11)$$

W drugim z równań układu (7.11) symbolem  $\mathbf{u}$  oznaczyliśmy siły (momenty) sterujące wywierane przez układy napędowe. Model (7.11) nosi nazwę pełnego modelu dynamiki manipulatora o elastycznych przegubach. Jeżeli przyjąć, że macierz  $\mathbf{S}(\mathbf{q}^{1n-1})$  formy energii kinetycznej (7.5) jest stała, model (7.11) upraszcza się do postaci

$$\begin{cases} \mathbf{Q}(\mathbf{q}^1) \ddot{\mathbf{q}}^1 + \mathbf{S}\ddot{\mathbf{q}}^2 + \mathbf{C}(\mathbf{q}^1, \dot{\mathbf{q}}^1) \dot{\mathbf{q}}^1 + \mathbf{D}(\mathbf{q}^1) + \mathbf{K}(\mathbf{q}^1 - \mathbf{q}^2) = \mathbf{0} \\ \mathbf{I}\ddot{\mathbf{q}}^2 + \mathbf{S}^T \ddot{\mathbf{q}}^1 - \mathbf{K}(\mathbf{q}^1 - \mathbf{q}^2) = \mathbf{u}, \end{cases} \quad (7.12)$$

która posiada własność linearyzowalności przez dynamiczne sprzężenie zwrotne. Najczęściej spotykany w literaturze jest model (7.12) z macierzą  $\mathbf{S} = \mathbf{0}$ , nazywany zredukowanym modelem dynamiki manipulatora o elastycznych przegubach. Model zredukowany

$$\begin{cases} \mathbf{Q}(\mathbf{q}^1) \ddot{\mathbf{q}}^1 + \mathbf{B}(\mathbf{q}^1, \dot{\mathbf{q}}^1) + \mathbf{K}(\mathbf{q}^1 - \mathbf{q}^2) = \mathbf{0} \\ \mathbf{I}\ddot{\mathbf{q}}^2 + \mathbf{K}(\mathbf{q}^2 - \mathbf{q}^1) = \mathbf{u}, \end{cases} \quad (7.13)$$

w którym  $\mathbf{B}(\mathbf{q}^1, \dot{\mathbf{q}}^1) = \mathbf{C}(\mathbf{q}^1, \dot{\mathbf{q}}^1) \dot{\mathbf{q}}^1 + \mathbf{D}(\mathbf{q}^1)$ , będziemy traktować w dalszych rozważaniach jako obiekt sterowania, dla którego przedstawimy zestaw algorytmów sterowania.

### Przykład 7.1.1 (Manipulator *EDDA*)

Do badania własności prezentowanych algorytmów sterowania wykorzystamy model manipulatora *EDDA* z uwzględnieniem oddziaływań elastycznych w przegubach. Macierzowo-wektorowe równania dynamiki tego manipulatora mają postać (7.13), a elementy modelu dynamiki są dane jako<sup>†</sup>

$$\mathbf{Q}(\mathbf{q}^1) = \begin{bmatrix} 2.833 + 0.462 \cos q_2^1 & 0.24 + 0.231 \cos q_2^1 \\ 0.24 + 0.231 \cos q_2^1 & 0.2325 \end{bmatrix},$$

$$\mathbf{C}(\mathbf{q}^1, \dot{\mathbf{q}}^1) = \begin{bmatrix} -0.231 \dot{q}_2^1 \sin q_2^1 & -0.231 (\dot{q}_1^1 + \dot{q}_2^1) \sin q_2^1 \\ 0.231 \dot{q}_1^1 \sin q_2^1 & 0 \end{bmatrix},$$

<sup>†</sup>Macierz  $\mathbf{Q}(\mathbf{q}^1)$  jest macierzą bezwładności manipulatora bez uwzględnienia bezwładności wirników. Macierz  $\mathbf{I}$  opisuje bezwładność samych wirników. Zauważmy, że suma obu macierzy daje macierz bezwładności  $\mathbf{Q}(\mathbf{q})$  manipulatora sztywnego opisaną wzorem (5.21).

$$\mathbf{D}(\mathbf{q}^1) = \begin{pmatrix} 93.2 \cos q_1^1 + 7.55 \cos (q_1^1 + q_2^1) \\ 7.55 \cos (q_1^1 + q_2^1) \end{pmatrix},$$

$$\mathbf{K} = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix}, \quad \mathbf{I} = \begin{bmatrix} 0.267 & 0 \\ 0 & 7.5 \times 10^{-3} \end{bmatrix}.$$

Wektor  $\mathbf{q}^1 = (q_1^1, q_2^1)^T \in \mathbb{R}^2$  opisuje położenia ramion manipulatora *EDDA*, zaś  $\mathbf{q}^2 = (q_1^2, q_2^2)^T \in \mathbb{R}^2$  oznacza położenia silników napędzających poszczególne ogniwa manipulatora. ■

## 7.2 Algorytmy sterowania

Spośród algorytmów zapewniających śledzenie trajektorii w przypadku pełnej znajomości modelu manipulatora przedstawimy trzy algorytmy: algorytm linearyzacji za pomocą statycznego sprzężenia zwrotnego, algorytm całkowania wstecznego oraz algorytm Ortegi-Lorii.

### 7.2.1 Algorytm linearyzacji statycznej

Dla modelu manipulatora o elastycznych przegubach (7.13) zdefiniujemy współrzędne stanu

$$\mathbf{x}^1 = \mathbf{q}^1, \quad \mathbf{x}^2 = \dot{\mathbf{q}}^1, \quad \mathbf{x}^3 = \mathbf{q}^2, \quad \mathbf{x}^4 = \dot{\mathbf{q}}^2.$$

Równania dynamiki we współrzędnych przyjmują postać

$$\begin{cases} \dot{\mathbf{x}}^1 = \mathbf{x}^2 \\ \dot{\mathbf{x}}^2 = -\mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{B}(\mathbf{x}^1, \mathbf{x}^2) + \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} (\mathbf{x}^3 - \mathbf{x}^1) \\ \dot{\mathbf{x}}^3 = \mathbf{x}^4 \\ \dot{\mathbf{x}}^4 = -\mathbf{I}^{-1} \mathbf{K} (\mathbf{x}^3 - \mathbf{x}^1) + \mathbf{I}^{-1} \mathbf{u}. \end{cases} \quad (7.14)$$

Przy oznaczeniach

$$\mathbf{F}(\mathbf{x}^1, \mathbf{x}^2) = -\mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{B}(\mathbf{x}^1, \mathbf{x}^2) - \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{x}^1,$$

$$\mathbf{G}(\mathbf{x}^1, \mathbf{x}^3) = -\mathbf{I}^{-1} \mathbf{K} (\mathbf{x}^3 - \mathbf{x}^1),$$

układ równań (7.14) zapiszemy jako

$$\begin{cases} \dot{\mathbf{x}}^1 = \mathbf{x}^2 \\ \dot{\mathbf{x}}^2 = \mathbf{F}(\mathbf{x}^1, \mathbf{x}^2) + \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{x}^3 \\ \dot{\mathbf{x}}^3 = \mathbf{x}^4 \\ \dot{\mathbf{x}}^4 = \mathbf{G}(\mathbf{x}^1, \mathbf{x}^3) + \mathbf{I}^{-1} \mathbf{u}. \end{cases}$$

Zastosujemy teraz transformację współrzędnych stanu

$$\boldsymbol{\xi} = (\xi^1, \xi^2, \xi^3, \xi^4)^T = \boldsymbol{\varphi}(\mathbf{x}) = \boldsymbol{\varphi}(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3, \mathbf{x}^4),$$

taką że

$$\begin{cases} \xi^1 = \mathbf{x}^1 \\ \xi^2 = \mathbf{x}^2 \\ \xi^3 = \mathbf{F}(\mathbf{x}^1, \mathbf{x}^2) + \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{x}^3 \\ \xi^4 = \frac{\partial \mathbf{F}(\mathbf{x}^1, \mathbf{x}^2)}{\partial \mathbf{x}^1} \mathbf{x}^2 + \frac{\partial \mathbf{F}(\mathbf{x}^1, \mathbf{x}^2)}{\partial \mathbf{x}^2} (\mathbf{F}(\mathbf{x}^1, \mathbf{x}^2) + \\ + \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{x}^3) + \frac{\partial}{\partial \mathbf{x}^1} (\mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{x}^3) \mathbf{x}^2 + \\ + \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{x}^4 = \mathbf{H}_1(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3) + \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{x}^4. \end{cases}$$

Ze względu na nieosobliwość macierzy  $\mathbf{Q}(\mathbf{x}^1)$  oraz  $\mathbf{K}$ ,  $\boldsymbol{\varphi}(\mathbf{x})$  jest globalnym dyfeomorfizmem przestrzeni stanu. Ostatecznie, równania dynamiki manipulatora w nowych współrzędnych zapiszemy w następującej formie

$$\begin{cases} \dot{\xi}^1 = \xi^2 \\ \dot{\xi}^2 = \xi^3 \\ \dot{\xi}^3 = \xi^4 \\ \dot{\xi}^4 = \mathbf{H}_2(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3, \mathbf{x}^4) + \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{I}^{-1} \mathbf{u}, \end{cases} \quad (7.15)$$

przy oznaczeniu

$$\begin{aligned} \mathbf{H}_2(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3, \mathbf{x}^4) = & \frac{\partial \mathbf{H}_1(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3)}{\partial \mathbf{x}^1} \mathbf{x}^2 + \frac{\partial \mathbf{H}_1(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3)}{\partial \mathbf{x}^2} (\mathbf{F}(\mathbf{x}^1, \mathbf{x}^2) + \\ & + \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{x}^3) + \frac{\partial \mathbf{H}_1(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3)}{\partial \mathbf{x}^3} \mathbf{x}^4 + \frac{\partial}{\partial \mathbf{x}^1} (\mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{x}^4) \mathbf{x}^2. \end{aligned}$$

Widać, że przy pomocy dyfeomorfizmu  $\boldsymbol{\varphi}(\mathbf{x})$  dynamika układu została częściowo zlinearyzowana. Aby uzyskać całkowitą linearyzację, niezbędne jest



zastosowanie statycznego sprzężenia zwrotnego

$$\mathbf{u} = \boldsymbol{\eta}(\mathbf{x}) + \boldsymbol{\psi}(\mathbf{x})\mathbf{v}, \quad (7.16)$$

gdzie  $\mathbf{v}$  oznacza nowe sterowanie układu, zaś

$$\begin{cases} \boldsymbol{\eta}(\mathbf{x}) = -\mathbf{IK}^{-1}\mathbf{Q}(\mathbf{x}^1) \mathbf{H}_2(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3, \mathbf{x}^4) \\ \boldsymbol{\psi}(\mathbf{x}) = \mathbf{IK}^{-1}\mathbf{Q}(\mathbf{x}^1). \end{cases} \quad (7.17)$$

Nietrudno sprawdzić, że sprzężenie zwrotne (7.16), (7.17)

$$\mathbf{u} = -\mathbf{IK}^{-1}\mathbf{Q}(\mathbf{x}^1) \mathbf{H}_2(\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3, \mathbf{x}^4) + \mathbf{IK}^{-1}\mathbf{Q}(\mathbf{x}^1) \mathbf{v} \quad (7.18)$$

przekształca układ (7.15) do postaci liniowej

$$\begin{cases} \dot{\xi}^1 = \xi^2 \\ \dot{\xi}^2 = \xi^3 \\ \dot{\xi}^3 = \xi^4 \\ \dot{\xi}^4 = \mathbf{v}. \end{cases} \quad (7.19)$$

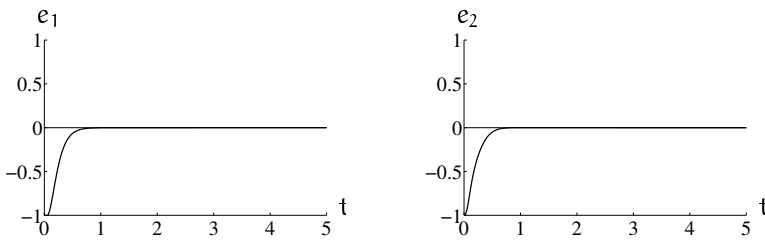
Po przeprowadzeniu linearyzacji modelu dynamiki manipulatora zadanie śledzenia zadanej trajektorii sprowadza się do śledzenia zadanej trajektorii przegubowej  $\xi_d^1(t)$  w układzie liniowym (7.19). Sterowanie  $\mathbf{v}$ , które zastosowane do układu (7.19) zapewni śledzenie zadanej trajektorii  $\xi_d^1(t)$ , tzn. doprowadzi do tego, że trajektoria przegubowa  $\xi^1(t)$  będzie dążyć asymptotycznie do  $\xi_d^1(t)$ , uzyskujemy metodami liniowej teorii regulacji. W tym celu założymy, że

$$\mathbf{v} = \xi_d^{1(4)} - \mathbf{R}_3\mathbf{e}^{(3)} - \mathbf{R}_2\mathbf{e}^{(2)} - \mathbf{R}_1\mathbf{e}^{(1)} - \mathbf{R}_0\mathbf{e}, \quad (7.20)$$

gdzie  $\mathbf{e}(t) = \xi^1(t) - \xi_d^1(t)$  oznacza błąd śledzenia trajektorii wyrażony we współrzędnych  $\xi$ . Błąd śledzenia w układzie (7.19) ze sterowaniem (7.20) jest opisany równaniem

$$\mathbf{e}^{(4)} + \mathbf{R}_3\mathbf{e}^{(3)} + \mathbf{R}_2\mathbf{e}^{(2)} + \mathbf{R}_1\mathbf{e}^{(1)} + \mathbf{R}_0\mathbf{e} = \mathbf{0}, \quad (7.21)$$

zaś warunek śledzenia  $\mathbf{e}(t) \rightarrow \mathbf{0}$  przy  $t \rightarrow +\infty$  można zapewnić poprzez taki dobór macierzy wzmocnień  $\mathbf{R}_3$ ,  $\mathbf{R}_2$ ,  $\mathbf{R}_1$ ,  $\mathbf{R}_0$ , aby układ (7.21) był eksponencjalnie stabilny. Właściwy dobór tych macierzy może zapewnić, oprócz eksponencjalnej stabilności, także pożądany przebieg błędu śledzenia (tłumienie krytyczne, minimalne przeregulowanie, itp.).



**Rysunek 7.2** Przebieg błędów śledzenia przy zastosowaniu algorytmu linearyzacji statycznej.

### Przykład 7.2.1 (Manipulator *EDDA*)

Działanie algorytmu linearyzacji statycznej zilustrujemy na przykładzie modelu manipulatora *EDDA* opisanego w przykładzie 7.1.1. Do obliczeń przyjęliśmy, że współczynniki elastyczności przegubów są równe  $k_1 = k_2 = 1000$ , co pozwoliło zaobserwować efekty związane z obecnością sił elastyczności. Trajektoria przegubowa  $\mathbf{q}_d^1(t)$  została zadana jako skok jednostkowy. Na rysunku 7.2 przedstawiono przykładowe przebiegi błędów śledzenia trajektorii dla parametrów układu sterowania równych odpowiednio  $\mathbf{R}_0 = \text{diag}\{10\}$ ,  $\mathbf{R}_1 = \text{diag}\{10\}$ ,  $\mathbf{R}_2 = \text{diag}\{50\}$ ,  $\mathbf{R}_3 = \text{diag}\{50\}$ . Poprzez wybór różnych wartości parametrów  $\mathbf{R}_i$  można kształtować charakter przebiegu błędów śledzenia trajektorii w poszczególnych przegubach. ■

### 7.2.2 Algorytm całkowania wstecznego

Rozważmy model manipulatora o elastycznych przegubach opisany równaniami (7.13). Zadaniem algorytmu sterowania jest zapewnienie śledzenia zadanej trajektorii przegubowej manipulatora. Zauważmy, że o ile zadana trajektoria przegubowa manipulatora ( $\mathbf{q}_d^1(t)$ ) jest dowolna, to zadana trajektoria silników ( $\mathbf{q}_d^2(t)$ ) wynika bezpośrednio z pierwszego równania (7.13)

$$\mathbf{q}_d^2 = \mathbf{q}_d^1 + \mathbf{K}^{-1}(\mathbf{Q}(\mathbf{q}_d^1) \ddot{\mathbf{q}}_d^1 + \mathbf{B}(\mathbf{q}_d^1, \dot{\mathbf{q}}_d^1)). \quad (7.22)$$

Zdefiniujmy współrzędne

$$\mathbf{x}^1 = \mathbf{q}^1, \quad \mathbf{x}^2 = \dot{\mathbf{q}}^1, \quad \mathbf{x}^3 = \mathbf{q}^2, \quad \mathbf{x}^4 = \dot{\mathbf{q}}^2,$$

w taki sposób, że

$$\mathbf{x}_d^1 = \mathbf{q}_d^1, \quad \mathbf{x}_d^2 = \dot{\mathbf{q}}_d^1, \quad \mathbf{x}_d^3 = \mathbf{q}_d^2, \quad \mathbf{x}_d^4 = \dot{\mathbf{q}}_d^2,$$

i przepiszymy równania (7.13) jako

$$\begin{cases} \dot{\mathbf{x}}^1 = \mathbf{x}^2 \\ \dot{\mathbf{x}}^2 = \mathbf{F}_1(\mathbf{x}^1, \mathbf{x}^2) + \mathbf{G}_1(\mathbf{x}^1) \mathbf{x}^3 \\ \dot{\mathbf{x}}^3 = \mathbf{x}^4 \\ \dot{\mathbf{x}}^4 = \mathbf{F}_2(\mathbf{x}^1, \mathbf{x}^3) + \mathbf{G}_2 \mathbf{u}, \end{cases} \quad (7.23)$$

gdzie  $\mathbf{F}_1(\mathbf{x}^1, \mathbf{x}^2) = -\mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{B}(\mathbf{x}^1, \mathbf{x}^2) - \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K} \mathbf{x}^1$ ,  $\mathbf{G}_1(\mathbf{x}^1) = \mathbf{Q}^{-1}(\mathbf{x}^1) \mathbf{K}$ ,  $\mathbf{F}_2(\mathbf{x}^1, \mathbf{x}^3) = -\mathbf{I}^{-1} \mathbf{K} (\mathbf{x}^3 - \mathbf{x}^1)$ ,  $\mathbf{G}_2 = \mathbf{I}^{-1}$ . Równania standardowe (7.23) przedstawimy w postaci zależnej od błędów  $\mathbf{e}^i = \mathbf{x}^i - \mathbf{x}_d^i$ ,  $i = 1, \dots, 4$

$$\begin{cases} \dot{\mathbf{e}}^1 = \mathbf{e}^2 \\ \dot{\mathbf{e}}^2 = \mathbf{F}_{12}(t, \mathbf{e}^1, \mathbf{e}^2) + \mathbf{G}_1(t, \mathbf{e}^1) \mathbf{e}^3 \\ \dot{\mathbf{e}}^3 = \mathbf{e}^4 \\ \dot{\mathbf{e}}^4 = \mathbf{F}_{22}(t, \mathbf{e}^1, \mathbf{e}^3) + \mathbf{G}_2 \mathbf{u}, \end{cases} \quad (7.24)$$

przy czym  $\mathbf{F}_{12}(t, \mathbf{e}^1, \mathbf{e}^2) = \mathbf{F}_1(t, \mathbf{e}^1, \mathbf{e}^2) + \mathbf{G}_1(t, \mathbf{e}^1) \mathbf{x}_d^3 - \dot{\mathbf{x}}_d^2$ ,  $\mathbf{G}_1(t, \mathbf{e}^1) = \mathbf{Q}^{-1}(t, \mathbf{e}^1) \mathbf{K}$ ,  $\mathbf{F}_{22}(t, \mathbf{e}^1, \mathbf{e}^3) = \mathbf{F}_2(t, \mathbf{e}^1, \mathbf{e}^3) - \dot{\mathbf{x}}_d^4$ ,  $\mathbf{G}_2 = \mathbf{I}^{-1}$ . Zauważmy, że równania (7.24) mają strukturę kaskadową, która pozwala na zastosowanie algorytmu sterowania typu *całkowania wstecznego*\*

Rozpocznijmy od rozważenia równania pierwszego podsystemu, a mianowicie

$$\dot{\mathbf{e}}^1 = \mathbf{e}^2. \quad (7.25)$$

Powyższy układ jest układem liniowym, w którym  $\mathbf{e}^2$  można potraktować jako sterowanie. Wybierzmy funkcję Lapunowa postaci

$$V_1(\mathbf{e}^1) = \frac{1}{2} \mathbf{e}^{1\top} \mathbf{e}^1. \quad (7.26)$$

Skoro zmienna  $\mathbf{e}^2$  jest sterowaniem układu (7.25), zastosujemy sprzężenie zwrotne

$$\mathbf{e}^2 = -\mathbf{R}_0 \mathbf{e}^1, \quad (7.27)$$

z macierzą  $\mathbf{R}_0 = \mathbf{R}_0^\top > 0$ . Wówczas, pochodna funkcji Lapunowa wzdłuż trajektorii układu (7.25)

$$\dot{V}_1 = -\mathbf{e}^{1\top} \mathbf{R}_0 \mathbf{e}^1 = -W_1(\mathbf{e}^1) \leq 0. \quad (7.28)$$

\*Termin ten proponujemy jako odpowiednik angielskiego zwrotu *integrator backstepping*.

Z teorii Lapunowa wynika, że wybór  $\mathbf{e}^2$  w taki sposób, aby spełnione było równanie (7.27), gwarantuje zbieżność błędu śledzenia  $\mathbf{e}^1(t)$  do  $\mathbf{0}$ . Łatwo można zauważyć, że błąd  $\mathbf{e}^1$  dąży do  $\mathbf{0}$  eksponencjalnie.

Następny krok całkowania wstecznego polega na rozważeniu drugiego, większego podsystemu układu (7.24), zawierającego dwa równania

$$\begin{cases} \dot{\mathbf{e}}^1 = \mathbf{e}^2 \\ \dot{\mathbf{e}}^2 = \mathbf{F}_{12}(t, \mathbf{e}^1, \mathbf{e}^2) + \mathbf{G}_1(t, \mathbf{e}^1) \mathbf{e}^3. \end{cases} \quad (7.29)$$

W tym podsystemie rolę sterowania będzie spełniać zmienna  $\mathbf{e}^3$ . Dla układu (7.29) wybierzmy funkcję Lapunowa

$$V_2(\mathbf{e}^1, \mathbf{e}^2) = V_1(\mathbf{e}^1) + \frac{1}{2}(\mathbf{e}^2 + \mathbf{R}_0\mathbf{e}^1)^\top (\mathbf{e}^2 + \mathbf{R}_0\mathbf{e}^1), \quad (7.30)$$

k której pochodna wzdłuż trajektorii układu (7.29) jest równa

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + (\mathbf{e}^2 + \mathbf{R}_0\mathbf{e}^1)^\top (\dot{\mathbf{e}}^2 + \mathbf{R}_0\dot{\mathbf{e}}^1) = \\ &= \dot{V}_1 + (\mathbf{e}^2 + \mathbf{R}_0\mathbf{e}^1)^\top (\mathbf{F}_{12}(t, \mathbf{e}^1, \mathbf{e}^2) + \mathbf{G}_1(t, \mathbf{e}^1) \mathbf{e}^3 + \mathbf{R}_0\mathbf{e}^2). \end{aligned} \quad (7.31)$$

Zażądamy teraz, aby

$$\mathbf{F}_{12} + \mathbf{G}_1\mathbf{e}^3 + \mathbf{R}_0\mathbf{e}^2 = -\mathbf{R}_1(\mathbf{e}^2 + \mathbf{R}_0\mathbf{e}^1),$$

gdzie  $\mathbf{R}_1 = \mathbf{R}_1^\top > \mathbf{0}$ , skąd wyliczamy sterowanie

$$\mathbf{e}^3 = -\mathbf{G}_1^{-1}(\mathbf{F}_{12} + (\mathbf{R}_1 + \mathbf{R}_0)\mathbf{e}^2 + \mathbf{R}_1\mathbf{R}_0\mathbf{e}^1) = -\mathbf{H}_3(t, \mathbf{e}^1, \mathbf{e}^2). \quad (7.32)$$

Widać, że dla tak wybranego sterowania funkcja Lapunowa ma pochodną równą

$$\begin{aligned} \dot{V}_2 &= -W_1(\mathbf{e}^1) - (\mathbf{e}^2 + \mathbf{R}_0\mathbf{e}^1)^\top \mathbf{R}_1(\mathbf{e}^2 + \mathbf{R}_0\mathbf{e}^1) = \\ &= -W_2(\mathbf{e}^1, \mathbf{e}^2) \leq 0. \end{aligned} \quad (7.33)$$

Na podstawie twierdzenia La Salle'a-Yoshizawy<sup>†</sup> można łatwo wywnioskować, że  $\mathbf{e}^3 = -\mathbf{H}_3(t, \mathbf{e}^1, \mathbf{e}^2)$  zapewnia śledzenie trajektorii zadanych  $\mathbf{x}_d^1, \mathbf{x}_d^2$  przez podsystem (7.29).

<sup>†</sup>Zobacz dodatek B.

W trzecim kroku algorytmu całkowania wstecznego rozważmy podsystem

$$\begin{cases} \dot{\mathbf{e}}^1 = \mathbf{e}^2 \\ \dot{\mathbf{e}}^2 = \mathbf{F}_{12}(t, \mathbf{e}^1, \mathbf{e}^2) + \mathbf{G}_1(t, \mathbf{e}^1) \mathbf{e}^3 \\ \dot{\mathbf{e}}^3 = \mathbf{e}^4, \end{cases} \quad (7.34)$$

przy czym zmienna  $\mathbf{e}^4$  spełnia rolę sterowania. Sterowanie stabilizujące dla układu (7.34) otrzymamy wybierając zależną od czasu funkcję Lapunowa

$$V_3(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3) = V_2(\mathbf{e}^1, \mathbf{e}^2) + \frac{1}{2}(\mathbf{e}^3 + \mathbf{H}_3(t, \mathbf{e}^1, \mathbf{e}^2))^T (\mathbf{e}^3 + \mathbf{H}_3(t, \mathbf{e}^1, \mathbf{e}^2)). \quad (7.35)$$

Pochodna  $\dot{V}_3$  funkcji Lapunowa wzdłuż trajektorii układu (7.34) jest równa

$$\dot{V}_3 = \dot{V}_2 + (\mathbf{e}^3 + \mathbf{H}_3(t, \mathbf{e}^1, \mathbf{e}^2))^T (\mathbf{e}^4 + \dot{\mathbf{H}}_3(t, \mathbf{e}^1, \mathbf{e}^2)). \quad (7.36)$$

Aby powyższa pochodna była niedodatnia, zastosujemy podstawienie

$$\mathbf{e}^4 + \dot{\mathbf{H}}_3(t, \mathbf{e}^1, \mathbf{e}^2) = -\mathbf{R}_2 (\mathbf{e}^3 + \mathbf{H}_3(t, \mathbf{e}^1, \mathbf{e}^2))$$

z macierzą  $\mathbf{R}_2$  symetryczną i dodatnio określoną. Stąd

$$\mathbf{e}^4 = -\dot{\mathbf{H}}_3(t, \mathbf{e}^1, \mathbf{e}^2) - \mathbf{R}_2 (\mathbf{e}^3 + \mathbf{H}_3(t, \mathbf{e}^1, \mathbf{e}^2)) = -\mathbf{H}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3). \quad (7.37)$$

Z definicji (7.37) oraz (7.36) mamy

$$\begin{aligned} \dot{V}_3 &= -W_2(\mathbf{e}^1, \mathbf{e}^2) - (\mathbf{e}^3 + \mathbf{H}_3(t, \mathbf{e}^1, \mathbf{e}^2))^T \mathbf{R}_2 \\ &\quad (\mathbf{e}^3 + \mathbf{H}_3(t, \mathbf{e}^1, \mathbf{e}^2)) = -W_3(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3) \leq 0. \end{aligned} \quad (7.38)$$

Ostatecznie, rozważmy cały układ (7.24) ze sterowaniem  $\mathbf{u}$ . Analogicznie do poprzednich kroków, sterowanie  $\mathbf{u}$  wyznaczymy poprzez analizę funkcji Lapunowa, w tym przypadku

$$V_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3, \mathbf{e}^4) = V_3(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3) + \frac{1}{2}(\mathbf{e}^4 + \mathbf{H}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3))^T (\mathbf{e}^4 + \mathbf{H}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3)). \quad (7.39)$$

Pochodna względem czasu funkcji  $V_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3, \mathbf{e}^4)$  opisanej powyższym

równaniem, liczona wzdłuż trajektorii układu (7.24), jest równa

$$\dot{V}_4 = -W_3(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3) + (\mathbf{e}^4 + \mathbf{H}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3))^T \left( \mathbf{F}_{22}(t, \mathbf{e}^1, \mathbf{e}^3) + \mathbf{G}_2 \mathbf{u} + \dot{\mathbf{H}}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3) \right). \quad (7.40)$$

Aby była ona niedodatnia, wystarczy spełnienie zależności

$$\begin{aligned} \mathbf{F}_{22}(t, \mathbf{e}^1, \mathbf{e}^3) + \mathbf{G}_2 \mathbf{u} + \dot{\mathbf{H}}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3) = \\ = -\mathbf{R}_3 (\mathbf{e}^4 + \mathbf{H}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3)) \end{aligned}$$

dla pewnej macierzy  $\mathbf{R}_3$  symetrycznej i dodatnio określonej. Z ostatniej zależności otrzymujemy algorytm sterowania  $\mathbf{u}$  zapewniający asymptotyczną stabilność układu (7.24), a mianowicie

$$\begin{aligned} \mathbf{u} = -\mathbf{G}_2^{-1} \left( \mathbf{F}_{22}(t, \mathbf{e}^1, \mathbf{e}^3) + \dot{\mathbf{H}}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3) + \right. \\ \left. + \mathbf{R}_3 (\mathbf{e}^4 + \mathbf{H}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3)) \right). \quad (7.41) \end{aligned}$$

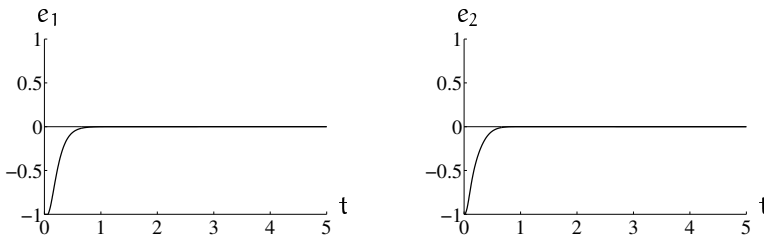
Po podstawieniu (7.41) do (7.40) dostaniemy

$$\begin{aligned} \dot{V}_4 = -W_3(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3) - (\mathbf{e}^4 + \mathbf{H}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3))^T \mathbf{R}_3 \\ (\mathbf{e}^4 + \mathbf{H}_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3)) = -W_4(t, \mathbf{e}^1, \mathbf{e}^2, \mathbf{e}^3, \mathbf{e}^4) \leq 0. \quad (7.42) \end{aligned}$$

Dowód stabilności algorytmu sterowania (7.41) zastosowanego do układu sterowania (7.24) opiera się na twierdzeniu La Salle'a-Yoshizawy lub na lemacie Barbalata.

### Przykład 7.2.2 (Manipulator *EDDA*)

Zastosujemy teraz algorytm całkowania wstecznego do modelu manipulatora *EDDA*. Do obliczeń przyjęto wartości  $k_1 = k_2 = 1000$  współczynników elastyczności przegubów, co pozwoliło zaobserwować efekty związane z obecnością sił elastyczności. Zadaną trajektorię przegubową  $\mathbf{q}_d^1(t)$  określiliśmy jako skok jednostkowy. Na rysunku 7.3 przedstawiono przykładowe przebiegi błędów śledzenia trajektorii w obu przegubach dla parametrów układu sterowania  $\mathbf{R}_0 = \text{diag}\{10\}$ ,  $\mathbf{R}_1 = \text{diag}\{10\}$ ,  $\mathbf{R}_2 = \text{diag}\{50\}$ ,  $\mathbf{R}_3 = \text{diag}\{50\}$ . Wybór innych parametrów  $\mathbf{R}_i$  pozwala uzyskać większe tłumienie błędów oraz szybszą zbieżność, ale powoduje wystąpienie przeregulowań. ■



Rysunek 7.3 Przebieg błędów śledzenia przy zastosowaniu algorytmu całkowania wstecznego.

### 7.2.3 Algorytm Ortegi-Lorii

Rozważmy teraz model manipulatora o elastycznych przegubach opisany równaniami (7.13), z algorytmem sterowania Ortegi-Lorii

$$\begin{cases} \mathbf{u} = \mathbf{I}\ddot{\mathbf{q}}_d^2 + \mathbf{K}(\mathbf{q}_d^2 - \mathbf{q}_d^1) - \mathbf{K}_{p2}\mathbf{e}^2 - \mathbf{K}_{d2}\dot{\boldsymbol{\theta}}^2 \\ \dot{\boldsymbol{\theta}}^i + \mathbf{M}_i\boldsymbol{\theta}^i = \mathbf{N}_i\dot{\mathbf{e}}^i, \quad i = 1, 2, \end{cases} \quad (7.43)$$

którego elementy składowe są zdefiniowane jak następuje:

$$\mathbf{q}_d^2 = \mathbf{K}^{-1}(\mathbf{Q}(\mathbf{q}^1)\ddot{\mathbf{q}}_d^1 + \mathbf{B}(\mathbf{q}^1, \dot{\mathbf{q}}_d^1) - \mathbf{K}_{p1}\mathbf{e}^1 - \mathbf{K}_{d1}\dot{\boldsymbol{\theta}}^1) + \mathbf{q}_d^1,$$

$$\mathbf{K}_{pi} = \text{diag}\{k_{pi}\}, \quad \mathbf{K}_{di} = \text{diag}\{k_{di}\}, \quad k_{pi}, k_{di} > 0,$$

$$\mathbf{M}_i = \text{diag}\{m_{ij}\}, \quad i = 1, 2, \quad j = 1, \dots, n,$$

$$\mathbf{N}_i = \text{diag}\{n_{ij}\}, \quad n_{ij} > \frac{\bar{\lambda}_{\hat{\mathbf{Q}}}}{\beta_2 \lambda_{\hat{\mathbf{Q}}}}, \quad 0 < \beta_2 < 1, \quad i = 1, 2, \quad j = 1, \dots, n,$$

$$\mathbf{e}^i = \mathbf{q}^i - \mathbf{q}_d^i, \quad i = 1, 2,$$

$$\hat{\mathbf{Q}} = \begin{bmatrix} \mathbf{Q}(\mathbf{q}^1) & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}.$$

Elementy macierzy  $\mathbf{K}_{pi}$ ,  $\mathbf{K}_{di}$ ,  $\mathbf{M}_i$ ,  $\mathbf{N}_i$ ,  $i = 1, 2$ , należy dobrać w taki sposób, aby zapewnić zbieżność algorytmu. W tym celu standardowo definiujemy  $\bar{\lambda}_{\hat{\mathbf{Q}}} = \max_{\mathbf{q}} \bar{\lambda}_{\hat{\mathbf{Q}}(\mathbf{q})}$ ,  $\lambda_{\hat{\mathbf{Q}}} = \min_{\mathbf{q}} \lambda_{\hat{\mathbf{Q}}(\mathbf{q})}$ . Zauważmy, że algorytm Ortegi-Lorii nie wymaga bezpośredniego pomiaru prędkości przegubów ani silników. Algorytm ten przypomina regulator PD, w którym sygnał  $\boldsymbol{\theta}^i$  podawany na wejście członu różniczkującego nie jest sygnałem błędu prędkości uzyskiwanym z bezpośredniego pomiaru, lecz przefiltrowanym sygnałem błędu położenia. Symbolem  $\boldsymbol{\theta}^1$  oznaczyliśmy estymowaną prędkość przegubów manipulatora, natomiast  $\boldsymbol{\theta}^2$  oznacza estymowaną prędkość silników napędzających poszczególne przeguby.

Równania układu zamkniętego złożonego z modelu (7.13) i algorytmu sterowania (7.43) można przedstawić w postaci

$$\begin{cases} \mathbf{Q}\ddot{\mathbf{e}} + (\mathbf{C} + \mathbf{C}_d)\dot{\mathbf{e}} + \mathbf{K}_p\mathbf{e} + \mathbf{K}_d\boldsymbol{\vartheta} = \mathbf{0} \\ \dot{\boldsymbol{\vartheta}} = -\mathbf{M}\boldsymbol{\vartheta} + \mathbf{N}\dot{\mathbf{e}}, \end{cases} \quad (7.44)$$

przy oznaczeniach  $\mathbf{e} = (\mathbf{e}^{1T}, \mathbf{e}^{2T})^T$ ,  $\boldsymbol{\vartheta} = (\boldsymbol{\vartheta}^{1T}, \boldsymbol{\vartheta}^{2T})^T$  oraz

$$\begin{aligned} \mathbf{C} &= \begin{bmatrix} \mathbf{C}(\mathbf{q}^1, \dot{\mathbf{q}}^1) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, & \mathbf{C}_d &= \begin{bmatrix} \mathbf{C}(\mathbf{q}^1, \dot{\mathbf{q}}_d^1) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \\ \mathbf{K}_p &= \begin{bmatrix} \mathbf{K}_{p1} + \mathbf{K} & -\mathbf{K} \\ -\mathbf{K} & \mathbf{K}_{p2} + \mathbf{K} \end{bmatrix}, & \mathbf{K}_d &= \begin{bmatrix} \mathbf{K}_{d1} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{d2} \end{bmatrix}, \\ \mathbf{M} &= \begin{bmatrix} \mathbf{M}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_2 \end{bmatrix}, & \mathbf{N} &= \begin{bmatrix} \mathbf{N}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{N}_2 \end{bmatrix}. \end{aligned}$$

Funkcją Lapunowa dla układu zamkniętego jest

$$V(\mathbf{x}, t) = \frac{1}{2}\dot{\mathbf{e}}^T\mathbf{Q}\dot{\mathbf{e}} + \frac{1}{2}\mathbf{e}^T\mathbf{K}_p\mathbf{e} + \frac{1}{2}\boldsymbol{\vartheta}^T\mathbf{K}_d\mathbf{N}^{-1}\boldsymbol{\vartheta} + \varepsilon\mathbf{e}^T\mathbf{Q}\dot{\mathbf{e}} - \varepsilon\boldsymbol{\vartheta}^T\mathbf{Q}\dot{\mathbf{e}}, \quad (7.45)$$

gdzie  $\mathbf{x} = (\mathbf{e}^T, \dot{\mathbf{e}}^T, \boldsymbol{\vartheta}^T)^T$ , a stała  $\varepsilon$  została wybrana w taki sposób, by zapewnić dodatnią określoność  $V$ . Obszar przyciągania trajektorii  $\mathbf{x}(t)$  układu zamkniętego może być zdefiniowany jako

$$\mathbf{A} = \{\mathbf{x} \in \mathbb{R}^{6n} \mid \|\mathbf{x}\| < c_2(\lambda_{\mathbf{N}})\},$$

przy czym

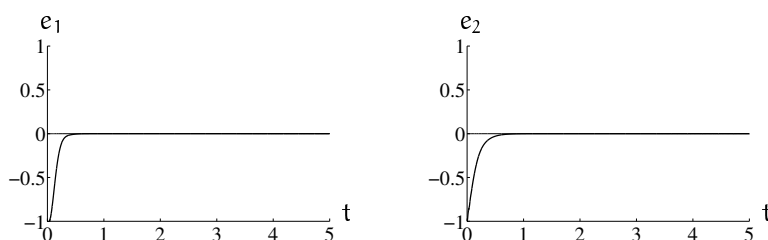
$$\lim_{\lambda_{\mathbf{N}} \rightarrow +\infty} c_2(\lambda_{\mathbf{N}}) = +\infty,$$

co oznacza półglobalną stabilność algorytmu Ortegi-Lorii.

### Przykład 7.2.3 (Manipulator *EDDA*)

Podobnie jak w przypadku poprzednich algorytmów, działanie algorytmu Ortegi-Lorii prześledzimy na przykładzie zadania śledzenia trajektorii przegubowej manipulatora *EDDA* mającej postać skoku jednostkowego. Ponownie przyjmijmy wartości współczynników elastyczności  $k_1 = k_2 = 1000$ . Na rysunku 7.4 przedstawiono przykładowe przebiegi błędów śledzenia trajektorii dla parametrów układu sterowania równych  $\mathbf{M}_i = \text{diag}\{100\}$ ,  $\mathbf{N}_i = \text{diag}\{400\}$ ,  $\mathbf{K}_{pi} = \text{diag}\{300\}$ ,  $\mathbf{K}_{di} = \text{diag}\{10\}$ ,  $i = 1, 2$ . Badania symulacyjne wykazały, że dla każdej wartości wzmocnienia regulatora  $\mathbf{K}_{pi}$  istnieje optymalna wartość  $\mathbf{K}_{di}$  zapewniająca największe tłumienie błędów śledzenia i nie powodująca przeregulowań. ■





Rysunek 7.4 Wykres błędów śledzenia przy zastosowaniu algorytmu Ortegi-Lorii.

Przeprowadzone symulacje pokazują, że wszystkie algorytmy przedstawione w tym rozdziale zapewniają praktycznie jednakową jakość śledzenia. Różnice między algorytmami są związane z ich warunkami implementacji. Największym stopniem skomplikowania cechuje się algorytm całkowania wstecznego, który dodatkowo wymaga pomiaru trzeciej pochodnej rzeczywistej trajektorii manipulatora. Najprostszy do implementacji jest algorytm Ortegi-Lorii, w którym dzięki zastosowaniu odpowiednich filtrów wymagany jest jedynie pomiar położenia przegubów i silników napędzających manipulator.

### 7.3 Komentarze i uwagi bibliograficzne

Model zredukowany dynamiki manipulatora o elastycznych przegubach można znaleźć w [CSB96]. Linearyzację przez dynamiczne sprzężenie zwrotne modelu dynamiki manipulatora o elastycznych przegubach przy stałej macierzy  $\mathbf{S}$  przeprowadzili De Luca i Lucibello [LL98]. Algorytm statycznej linearyzacji manipulatora o elastycznych przegubach został zaczerpnięty z [Vid92]. Idea całkowania wstecznego pochodzi z monografii [KKK95], natomiast zastosowanie algorytmu całkowania wstecznego do zadania sterowania manipulatorem o elastycznych przegubach przedstawiono w [TM98]. Algorytm Ortegi-Lorii pochodzi z pracy [LO95]. Algorytmy adaptacyjne dla manipulatora o elastycznych przegubach otrzymane poprzez zastosowanie metody osobliwych zaburzeń można znaleźć u Khorasaniego [Kho92]. Problemem otwartym pozostaje znalezienie algorytmu sterowania, gdy macierz  $\mathbf{S}(\mathbf{q}^{1n-1})$  nie jest stała, a także modeli dynamiki i algorytmów sterowania dla przypadku, gdy elastyczność przegubów nie pochodzi wyłącznie od sił sprężystości. Uwzględnienie elastyczności ramion powoduje znaczną komplikację modelu dynamiki manipulatora. Praktyczne aspekty mode-

lowania manipulatora o elastycznych ramionach za pomocą metody elementów skończonych można znaleźć w [BS95].

## Literatura

- [BS95] W. Beres i J. Sasiadek, Finite elements dynamic model of multilink flexible manipulators. *Appl. Mathematics and Computer Science*, 5(2):231–262, 1995.
- [CSB96] C. Canudas de Wit, B. Siciliano i G. Bastin, *Theory of Robot Control*. Springer-Verlag, New York, 1996.
- [Kho92] K. Khorasani, Adaptive control of flexible-joint robots. *IEEE Trans. Robotics Automat.*, 8(2):250–267, 1992.
- [KKK95] M. Krstić, I. Kanellakopoulos i P. Kokotović, *Nonlinear and Adaptive Control Design*. J. Wiley and Sons, New York, 1995.
- [LL98] A. De Luca i P. Lucibello, A general algorithm for dynamic feedback linearization of robots with elastic joints. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 504–510, Leuven, 1998.
- [LO95] A. Loria i R. Ortega, On tracking control of rigid and flexible joint robots. *Appl. Mathematics and Computer Science*, 5(2):329–342, 1995.
- [TM98] K. Tchoń i A. Mazur, A backstepping control algorithm for flexible joint robotic manipulators. W: *Proc. MMAR Symposium*, vol. 3, strony 891–896, Międzyzdroje, 1998.
- [Vid92] M. Vidyasagar, *Nonlinear Systems Analysis*. Prentice Hall, New Jersey, 1992.

Część III

**Algorytmy planowania ruchu  
i sterowania robotów  
mobilnych**



## Rozdział 8

# Zadanie planowania ruchu

Zgodnie z określeniem podanym w podrozdziale 2.2, nieholonomiczne układy robotyczne są układami spełniającymi nieholonomiczne, czyli niecałkowalne, ograniczenia fazowe. Układy, których równania ograniczeń (więzów) można scałkować noszą miano holonomicznych.

### 8.1 Pojęcia podstawowe

Podstawową własnością układów holonomicznych jest możliwość eliminacji ograniczeń przez zdefiniowanie układu na rozmaitości konfiguracyjnej, której kowymiar jest równy liczbie niezależnych ograniczeń. Tę własność zilustrujemy następującym przykładem.

#### Przykład 8.1.1 (Ograniczenia holonomiczne)

Niech będzie dany układ dwuwymiarowy opisany współrzędnymi uogólnionymi  $\mathbf{q} = (x, y)^T \in \mathbb{R}^2$ , podlegający ograniczeniom fazowym

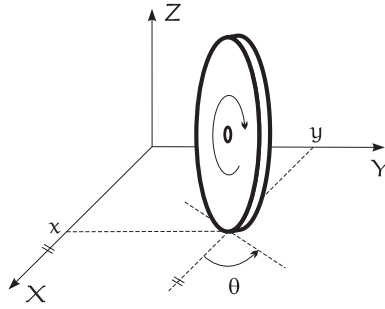
$$\mathbf{A}(x, y) \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = x\dot{x} + y\dot{y} = 0. \quad (8.1)$$

Z twierdzenia 2.2.1 wynika, że powyższe ograniczenia są holonomiczne, co łatwo wykazać bezpośrednio, bowiem istnieje funkcja

$$\Phi(x, y) = \frac{1}{2} (x^2 + y^2),$$

taka że

$$\frac{d\Phi}{dt} = \mathbf{A}(x, y) \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = 0.$$



Rysunek 8.1 Koło toczące się bez poślizgu.

Ograniczenie (8.1) definiuje w  $\mathbb{R}^2$  rozmiatość konfiguracyjną mającą postać okręgu, którego promień zależy od warunku początkowego  $(x_0, y_0)^T$ . We współrzędnych biegunowych

$$\begin{cases} x = r \cos \varphi \\ y = r \sin \varphi \end{cases}$$

rozmiatość ta jest opisana równaniem  $r = \text{const.}$  Ograniczenie (8.1) we współrzędnych  $\varphi$  nowej rozmiatości konfiguracyjnej nie jest aktywne, gdyż struktura tej rozmiatości zapewnia automatycznie spełnienie tego ograniczenia. ■

Dla kontrastu przedstawimy na przykładach dwa ważne źródła ograniczeń nieholonomicznych — toczenie się bez poślizgu kół robotów mobilnych i zasadę zachowania momentu pędu struktur szybujących.

### Przykład 8.1.2 (Koło toczące się bez poślizgu)

Ograniczenie nieholonomiczne dla koła poruszającego się po płaszczyźnie (przedstawionego na rysunku 8.1) o wektorze stanu  $\mathbf{q} = (x, y, \theta)^T$ , wynika z warunku braku poślizgu bocznego

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0. \quad (8.2)$$

Nieholonomiczność ograniczenia (8.2) udowodnimy wykazując, że nie istnieje funkcja  $\Phi(x, y, \theta)$ , taka że jej pochodna względem wektora stanu jest, z dokładnością do pewnej funkcji  $\alpha(x, y, \theta) \neq 0$ , równa lewej stronie (8.2), czyli

$$\left( \frac{\partial \Phi}{\partial x}, \frac{\partial \Phi}{\partial y}, \frac{\partial \Phi}{\partial \theta} \right) = (\alpha \sin \theta, -\alpha \cos \theta, 0). \quad (8.3)$$

Z warunku równości pochodnych mieszanych, uzyskanych z różniczkowania odpowiednich wyrażeń w (8.3), uzyskujemy układ równań

$$\begin{cases} 0 = \frac{\partial^2 \Phi}{\partial x \partial \theta} = \frac{\partial^2 \Phi}{\partial \theta \partial x} = \frac{\partial \alpha}{\partial \theta} \sin \theta + \alpha \cos \theta \\ 0 = \frac{\partial^2 \Phi}{\partial y \partial \theta} = \frac{\partial^2 \Phi}{\partial \theta \partial y} = -\frac{\partial \alpha}{\partial \theta} \cos \theta + \alpha \sin \theta, \end{cases}$$

którego rozwiązanie wymaga, by  $\alpha \equiv 0$ , co dowodzi, że nie istnieją funkcje  $\alpha$ ,  $\Phi$  spełniające (8.3), czyli ograniczenie (8.2) jest nieholonomiczne. Łatwiejszy dowód nieholonomiczności ograniczenia (8.2) polega na zauważeniu, że funkcja  $\Phi$  nie może zależeć od zmiennej  $\theta^*$ . Zatem, aby było spełnione pierwsze równanie w (8.3),  $\alpha(x, y, \theta) = \frac{\tilde{\alpha}(x, y)}{\sin \theta}$ , gdzie  $\tilde{\alpha}(x, y)$  jest dowolną funkcją  $x$  i  $y$ , co wyklucza spełnienie równania drugiego. ■

### Przykład 8.1.3 (Spadający kot)

Spadającego kota modelujemy jako układ dwóch ciał sztywnych połączonych przegubem o trzech stopniach swobody, oznaczonym na rysunku 8.2 literą P. Wprowadzamy układy współrzędnych: układ  $X_0 Y_0 Z_0$  jest układem ciała związanym ze środkiem masy kota, natomiast w układach  $X_1 Y_1 Z_1$ ,  $X_2 Y_2 Z_2$  są opisywane, odpowiednio, ciała sztywne związane z przednią i tylną częścią ciała kota. Parametry opisujące każde z ciał sztywnych to skumulowane masy  $m_1$ ,  $m_2$  oraz macierze bezwładności  $I_1$ ,  $I_2$ . Rozmaitość konfiguracyjną stanowią położenia i orientacje układów 1, 2 w układzie środka masy

$$\mathcal{Q} = \text{SE}(3) \times \text{SE}(3) \ni \left( \begin{bmatrix} \mathbf{R}_1 & \mathbf{T}_1 \\ \mathbf{0} & 1 \end{bmatrix}, \begin{bmatrix} \mathbf{R}_2 & \mathbf{T}_2 \\ \mathbf{0} & 1 \end{bmatrix} \right).$$

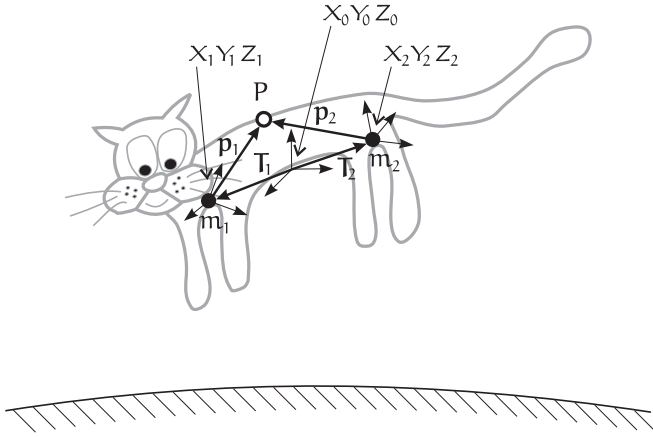
Na współrzędne konfiguracyjne są nałożone ograniczenia holonomiczne

$$m_1 \mathbf{T}_1 + m_2 \mathbf{T}_2 = \mathbf{0}, \quad \mathbf{T}_1 + \mathbf{R}_1 \mathbf{p}_1 = \mathbf{T}_2 + \mathbf{R}_2 \mathbf{p}_2,$$

a ograniczenia fazowe wynikają z zasady zachowania momentu pędu w układzie  $X_0 Y_0 Z_0$ . Przy założeniu, że w chwili początkowej moment pędu wynosi  $\mathbf{0}$ , ograniczenia fazowe są postaci

$$\mathbf{J}_1 \boldsymbol{\omega}_1 + m_1 \mathbf{T}_1 \times \dot{\mathbf{T}}_1 + \mathbf{J}_2 \boldsymbol{\omega}_2 + m_2 \mathbf{T}_2 \times \dot{\mathbf{T}}_2 = \mathbf{0}.$$

\*Zobacz trzecie równanie w (8.3).



Rysunek 8.2 Układy współrzędnych spadającego kota.

Prędkości kątowe w przestrzeni i macierze bezwładności w układzie  $X_0 Y_0 Z_0$  są zadane wzorami

$$\begin{aligned} \dot{\mathbf{R}}_1 &= [\boldsymbol{\omega}_1] \mathbf{R}_1, & \mathbf{J}_1 &= \mathbf{R}_1 \mathbf{I}_1 \mathbf{R}_1^T, \\ \dot{\mathbf{R}}_2 &= [\boldsymbol{\omega}_2] \mathbf{R}_2, & \mathbf{J}_2 &= \mathbf{R}_2 \mathbf{I}_2 \mathbf{R}_2^T. \end{aligned} \quad (8.4)$$

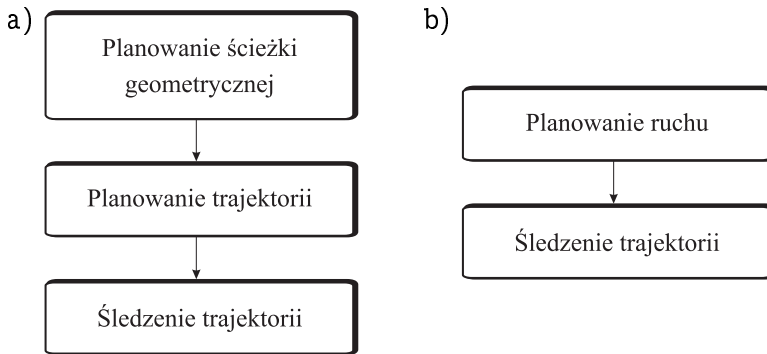
Biorąc pod uwagę ograniczenia konfiguracyjne, rozmaitość konfiguracyjną zawężamy do  $\mathcal{Q} = \text{SO}(3) \times \text{SO}(3)$ . Nietrudno sprawdzić, że ograniczenia fazowe przyjmą postać

$$\begin{bmatrix} \mathbf{R}_1 \mathbf{I}_1 \mathbf{R}_1^T + \delta \mathbf{T}_{12} [\mathbf{R}_1 \mathbf{p}_1] & \mathbf{R}_2 \mathbf{I}_2 \mathbf{R}_2^T - \delta \mathbf{T}_{12} [\mathbf{R}_2 \mathbf{p}_2] \end{bmatrix} \begin{pmatrix} \boldsymbol{\omega}_1 \\ \boldsymbol{\omega}_2 \end{pmatrix} = \mathbf{0}, \quad (8.5)$$

gdzie  $\delta = m_1 m_2 / (m_1 + m_2)$ , a  $\mathbf{T}_{12} = [\mathbf{R}_2 \mathbf{p}_2 - \mathbf{R}_1 \mathbf{p}_1] = [\mathbf{T}_1 - \mathbf{T}_2]$ . Zauważmy, że ograniczenie (8.5) nie jest w postaci Pfaffa. By nadać mu tę postać należy przyjąć pewną parametryzację macierzy obrotu  $\mathbf{R}_1, \mathbf{R}_2$ , np. przy pomocy kątów Eulera, a następnie, korzystając z zależności (8.4), wyliczyć zależność prędkości kątowych  $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2$  od pochodnych parametrów. Przyjęcie parametryzacji grupy obrotów faktycznie zmienia rozmaitość konfiguracyjną z  $\text{SO}(3) \times \text{SO}(3)$  na przestrzeń  $\mathbb{R}^3 \times \mathbb{R}^3 \cong \mathbb{R}^6$ , w której na konfiguracje układu są nałożone trzy ograniczenia nieholonomiczne. ■

Dla układu podlegającego ograniczeniom nieholonomicznym ograniczenia fazowe w każdym punkcie przestrzeni konfiguracyjnej określają dopuszczalne prędkości ruchu układu, nie zmniejszając w żaden sposób wielkości dostępnej przestrzeni konfiguracyjnej. Z tego powodu redukcja wymiaru

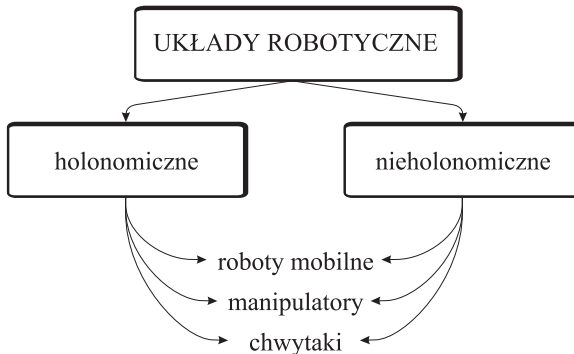




**Rysunek 8.3** Etapy realizacji zadań dla układu: a) holonomicznego, b) nieholonomicznego.

przestrzeni konfiguracyjnej nie jest możliwa. Rozważmy *zadanie planowania ruchu* układu robotycznego polegające na wyznaczeniu oddziaływań przeprowadzających układ od konfiguracji początkowej do zadanej konfiguracji końcowej. Jak wiadomo, planowanie ruchu dla układów holonomicznych odbywa się zwykle w trzech etapach (zobacz rysunek 8.3). Pierwszy etap to planowanie ścieżki, polegające na wyznaczeniu toru ruchu układu między stanem początkowym a docelowym. Kolejnym etapem jest planowanie trajektorii polegające na parametryzacji czasowej ścieżki geometrycznej, czyli określeniu prędkości ruchu wzdłuż ścieżki. Po dwóch zadaniach planowania następuje śledzenie zadanej trajektorii realizowane przez układ sterowania (śledzenia).

Ponieważ w przypadku układów nieholonomicznych charakterystyk pozycyjnych nie można rozpatrywać w oderwaniu od prędkościowych, dwa pierwsze etapy planowania muszą być realizowane równocześnie. W literaturze robotycznej zadanie wyznaczenia położeń i prędkości realizujących ruch między stanem początkowym a docelowym w sposób zgodny z ograniczeniami nieholonomicznymi jest określane mianem *zadania nieholonomicznego planowania ruchu*. Zarówno dla układów holonomicznych, jak i nieholonomicznych, zadania planowania są realizowane w trybie *off-line*, tj. trajektoria ruchu jest wyliczana w całości, zanim zostanie przesłana do układu sterowania w celu realizacji. Planowanie (ruchu, trajektorii, ścieżki) opiera się na wykorzystaniu wyidealizowanego modelu matematycznego układu robotycznego i jego otoczenia, i jest zwykle realizowane w układzie otwartym. Zamknięcie pętli sprzężenia zwrotnego jest konieczne przy śledzeniu zaplanowanej trajektorii w celu zapewnienia odporności al-



Rysunek 8.4 Podział układów robotycznych.

gorytmu sterowania na niedokładności modelu. Sterowanie w trybie czasu rzeczywistego nakłada ostre wymagania na złożoność obliczeniową zastosowanych algorytmów sterowania.

Z podziału układów robotycznych przedstawionego schematycznie na rysunku 8.4 wynika, że w zasadzie każdy układ robotyczny można zrealizować zarówno w wersji holonomicznej, jak i nieholonomicznej. Jednak w praktyce, większość manipulatorów jest holonomiczna, zaś roboty mobilne są zwykle układami nieholonomicznymi.

Zastanówmy się, jakie przesłanki podyktowane specyfiką układów nieholonomicznych przemawiają za realizacją układu robotycznego w wersji holonomicznej lub nieholonomicznej. Niektóre z nich zestawiliśmy poniżej.

- Sterowanie układami nieholonomicznymi jest istotnie trudniejsze od sterowania układami holonomicznymi, ze względu na mniejszą liczbę sterowań niż stopni swobody układu. Zatem, gdyby kryterium wyboru była prostota układu sterowania, preferowana byłaby wersja holonomiczna.
- Niektóre roboty są ze swej istoty nieholonomiczne, np. kosmiczne roboty szybujące, czy roboty podwodne. Ich nieholonomiczność wynika z prawa zachowania momentu pędu, generującego ograniczenia fazowe.
- W operacjach manipulacyjnych wykonywanych przez chwytki, zaletą chwytaków nieholonomicznych jest mała liczba silników sterujących. palcami chwytaków potrzebna do zapewnienia pełnych możliwości manipulacyjnych. Niezależnie od typu układu robotycznego, silniki napędzające chwytak mogą być umieszczone bądź w obrębie chwytaka, bądź z dala od niego. Umieszczenie silników w obrębie chwyt-

taka wpływa na wzrost masy i momentów bezwładności układu powodując zwiększenie kosztu (energii) sterowania. Z drugiej strony, jeśli silniki zostały umieszczone poza obrębem chwytaka, pojawiają się trudności z efektywnym przekazywaniem napędu do chwytaka. W konsekwencji, ograniczenie liczby silników w przypadku chwytaków nieholonomicznych wpływa korzystnie na własności dynamiczne manipulatora, ale zmusza do zastosowania bardziej złożonych algorytmów sterowania. Ponadto, manewry obiektem manipulowanym z wykorzystaniem chwytaka nieholonomicznego są bardziej zamaszyste, co jest szczególnie niekorzystne w środowisku kolizyjnym. Podsumowując powiemy, że gdy zadaniem robota jest częste manipulowanie obiektami, zalecane są chwytaki holonomiczne (często antropomorficzne lub hiperredundantne). Natomiast gdy manipulowanie odbywa się sporadycznie, warto rozważyć zastosowanie chwytaka nieholonomicznego poprawiającego własności dynamiczne manipulatora przy zachowaniu jego możliwości manipulacyjnych.

- Roboty mobilne, wyposażone w układy sensoryczne wykorzystywane przez algorytmy samolokalizacji i nawigacji, są często konstruowane jako obiekty holonomiczne o symetrycznym kształcie i ograniczonej prędkości ruchu<sup>†</sup>. W przyszłości, gdy roboty mobilne utracą symetrię przestrzenną i wzrosną ich prędkość, stanie się nieodzowne konstruowanie ich jako układów nieholonomicznych.
- Podejmowane próby projektowania manipulatorów w wersji nieholonomicznej mają, jak się wydaje, niewielkie szanse na realizację praktyczną.

Uznając odrębność i specyfikę metod planowania ruchu układów nieholonomicznych, dokonamy obecnie ich klasyfikacji. Jako pierwsze kryterium przyjmijmy zakres zastosowania metod. Wyróżnimy tu dwie klasy metod planowania ruchu:

- Metody globalne, w których cała trajektoria robota mobilnego jest planowana w procesie iteracyjnym. Ich działanie polega na minimalizacji pewnej funkcji błędu. Do wad metod globalnych należą: duża złożoność obliczeniowa oraz istotne utrudnienie zadania, w przypadku gdy niektóre położenia w przestrzeni konfiguracyjnej są zabronione

---

<sup>†</sup>Mogą bez trudu przeorientować się „w miejscu”, a ich kształt ułatwia planowanie ścieżki techniką powiększania przeszkód.

(np. z powodu występowania przeszkód). Metody globalne są mało odporne na zmiany warunków zadania planowania, jak np. dopuszczenie ruchomych przeszkód, umożliwiając jednak optymalizację globalnych kryteriów jakości.

- Metody lokalne, które na podstawie własności robota mobilnego jako układu sterowania opisanego równaniami kinematyki (2.145) wyznaczają sterowania wokół bieżącego punktu przestrzeni konfiguracyjnej, realizujące ruch w kierunku zmniejszającym odległość do celu. Rozwiązanie globalne powstaje przez sklejenie lokalnych fragmentów sterowań. Zaletą metod lokalnych jest mniejsza złożoność obliczeniowa niż metod globalnych (mogą być wykorzystywane w trybie czasu rzeczywistego). Główną wadą jest konieczność skojarzenia metody lokalnej z globalnym układem planowania wyznaczającym podcele dla metod lokalnych w przypadku istnienia przeszkód. Metody lokalne, choć umożliwiają optymalizację kryteriów lokalnych, nie gwarantują globalnej optymalizacji ruchu, gdyż decyzje lokalnie optymalne niekoniecznie prowadzą do najlepszego rozwiązania globalnego.

Ze względu na stopień uniwersalności, metody planowania ruchu można również podzielić na dwie grupy:

- Metody ogólnego przeznaczenia, stosowalne do szerokiej klasy układów sterowania, w tym także do układów nieholonomicznych. Do metod ogólnego przeznaczenia zaliczamy: metodę bazującą na Zasadzie Maksimum Pontriagina, metodę Newtona, metodę wykorzystującą technikę uśredniania oraz metodę Lie-algebraiczną.
- Metody specjalizowane, których zakres stosowalności jest ograniczony do podklasy układów nieholonomicznych o szczególnej strukturze. Do metod specjalizowanych zaliczamy: metodę sterowań sinusoidalnych, metodę osiągania podcelów, metodę Reedsa-Sheppa, metodę bazującą na twierdzeniu Stokesa, oraz metodę wykorzystującą własność nilpotentności.

Kolejny podział metod planowania jest podyktowany stopniem uwzględnienia dynamiki układu robotycznego podczas planowania ruchu. Z uwagi na to kryterium, metody planowania dzielimy na kinematyczne, które nie uwzględniają dynamiki układu nieholonomicznego oraz dynamiczne, które ją uwzględniają.

## 8.2 Technika nawiasu Liego

Z natury ograniczeń nieholonomicznych\* wynika, że pojęcie kinematyki układów nieholonomicznych jest szersze niż układów holonomicznych (manipulatorów). Dla tych ostatnich, zgodnie z formułą (2.40), za kinematykę uważa się przekształcenie między rozmaitościami konfiguracyjnymi (przebiegową i zadaniową), a więc obiektami opisującymi położenia, nie uwzględniające prędkości ruchu. Inaczej jest w przypadku układów nieholonomicznych, których kinematyka stanowi połączenie charakterystyk położeniowych i prędkościowych opisane przez układ sterowania (2.145)

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{q})u_i, \quad (8.6)$$

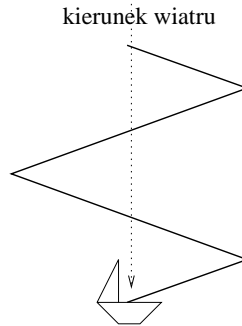
którego przestrzeń stanu pokrywa się z przestrzenią konfiguracyjną układu nieholonomicznego,  $\dim \mathbf{q} = n > m = \dim \mathbf{u}$ . Występujące w formule (8.6) pola wektorowe  $\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_m$  nazywamy *polami stowarzyszonymi* lub *generatorami* układu. Zakładamy, że sterowania dopuszczalne należą do przestrzeni funkcji całkowalnych z kwadratem  $\mathbb{L}_m^2[0, T]$ .

Pożądaną własnością każdego układu sterowania jest sterowalność, która oznacza, że istnieją sterowania dopuszczalne przeprowadzające stan układu z dowolnego stanu do dowolnego innego stanu w skończonym czasie. Dalej wymagać będziemy nieco mocniejszej własności, tzw. lokalnej sterowalności w krótkim czasie, która dla układu (8.6) polega na istnieniu sterowań dopuszczalnych przeprowadzających stan układu w każdym kierunku w przestrzeni fazowej w dowolnie krótkim czasie<sup>†</sup>.

Rozważmy zatem, jakie kierunki można wygenerować przy pomocy stałych sterowań  $\mathbf{u}$  w zadanym punkcie przestrzeni stanu  $\mathbf{q}_0$  układu (8.6). Włączając jedną wybraną składową wektora sterowań, a wyłączając składowe pozostałe, np. kładąc  $u_i = \pm 1$  oraz  $u_j = 0$  dla  $j \neq i$ ,  $i = 1, \dots, m$ , powodujemy ewolucję układu (8.6) wzdłuż trajektorii (krzywych całkowych) pól wektorowych — generatorów  $\mathbf{g}_i$ . Dzięki takiemu postępowaniu, stosując stałe sterowania, uzyskujemy możliwość poruszania się w obrębie podprzestrzeni  $\mathcal{P} = \text{span}_{\mathbb{R}}\{\mathbf{g}_1(\mathbf{q}_0), \dots, \mathbf{g}_m(\mathbf{q}_0)\}$  rozpiętej przez generatory układu w punkcie  $\mathbf{q}_0$ . Zauważmy, że nawet wtedy, gdy układ generatorów jest liniowo niezależny, liczba niezależnych kierunków wynosi  $m < n$ , co nie wystarcza żeby zapewnić sterowalność. Powstaje pytanie, jakie inne kierunki

\*Zobacz podrozdziały 2.2 i 2.4.

<sup>†</sup>Zobacz dodatek A.4.



**Rysunek 8.5** Kierunek wypadkowy realizowany przez złożenie kierunków prostych na przykładzie manewru halsowania.

można wygenerować przy pomocy sterowań  $\mathbf{u}$ ? Do opisu tych kierunków niezbędne jest wykorzystanie operacji *nawiasu Liego* wprowadzonej w rozdziale 2.2<sup>‡</sup>. Przypomnijmy, że we współrzędnych przestrzeni stanu nawias Liego jest zadany formułą (2.34)

$$[\mathbf{X}, \mathbf{Y}](\mathbf{q}) = \frac{\partial \mathbf{Y}}{\partial \mathbf{q}} \mathbf{X}(\mathbf{q}) - \frac{\partial \mathbf{X}}{\partial \mathbf{q}} \mathbf{Y}(\mathbf{q}). \quad (8.7)$$

Przykładem operacji nawiasu Liego jest iloczyn wektorowy w  $\mathbb{R}^3$ , zdefiniowany wzorem (2.1). W algebrze pól wektorowych operacja ta przypisuje parze pól  $\mathbf{X}, \mathbf{Y}$  pole wektorowe  $[\mathbf{X}, \mathbf{Y}]$ . Wybierając stan  $\mathbf{q} \in \mathbb{R}^n$  stwierdzamy, że wektor  $[\mathbf{X}, \mathbf{Y}](\mathbf{q})$  zwykle nie leży w płaszczyźnie wyznaczonej przez wektory  $\mathbf{X}(\mathbf{q}), \mathbf{Y}(\mathbf{q})$ , a zatem pole  $[\mathbf{X}, \mathbf{Y}]$  oferuje w punkcie  $\mathbf{q}$  dodatkowy kierunek ruchu, liniowo niezależny od kierunków wyznaczonych przez pola-argumenty. Przykładem praktycznej sytuacji, w której jest konieczne uzyskanie kierunku ruchu niemożliwego do realizacji przy pomocy pojedynczego sterowania, jest zadanie żeglowania pod wiatr. Halsowanie przedstawione na rysunku 8.5 jest sposobem realizacji takiego manewru.

Dla układu sterowania (8.6) wprowadzimy teraz pojęcie *algebry Liego* rozumianej jako najmniejsza przestrzeń liniowa pól wektorowych zawierająca generatory układu i zamknięta ze względu na nawias Liego zdefiniowany zależnością (8.7). Zamkniętość oznacza, że procedura generacji nawiasów Liego jest inicjowana generatorami układu (8.6), a następnie powtarzana rekurencyjnie dla wszystkich pól wektorowych uprzednio utworzonych. Algebra Liego w danym punkcie przestrzeni stanu opisuje zbiór dopuszczalnych kierunków ruchu układu.

<sup>‡</sup>Zobacz także dodatek A.3.

Związek kierunków ruchu ze sterowalnością podaje *twierdzenie Chow*<sup>S</sup>, które stanowi, że

**Twierdzenie 8.2.1** *Jeżeli w każdym punkcie przestrzeni stanu algebra Liego układu (8.6) rozpina przestrzeń stanu, to układ jest sterowalny w krótkim czasie.*

Poniżej, na kilku przykładach, zilustrujemy aspekty obliczeniowe wyznaczania algebry Liego wybranych układów nieholonomicznych.

### Przykład 8.2.1 (Jednokołowy robot mobilny)

Wyznamy algebrę Liego dla jednokołowego robota mobilnego opisanego równaniami

$$\dot{\mathbf{q}} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos \theta \\ \sin \theta \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u_2 = \mathbf{g}_1(\mathbf{q})u_1 + \mathbf{g}_2(\mathbf{q})u_2. \quad (8.8)$$

Korzystając z definicji (8.7) wyliczamy kilka nawiasów Liego

$$\begin{aligned} [\mathbf{g}_1, \mathbf{g}_1] &= \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, & [\mathbf{g}_1, \mathbf{g}_2] &= \begin{pmatrix} \sin \theta \\ -\cos \theta \\ 0 \end{pmatrix}, \\ [\mathbf{g}_1, [\mathbf{g}_1, \mathbf{g}_2]] &= \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, & [\mathbf{g}_2, [\mathbf{g}_1, \mathbf{g}_2]] &= \begin{pmatrix} \cos \theta \\ \sin \theta \\ 0 \end{pmatrix} = \mathbf{g}_1. \end{aligned}$$

Algebra Liego układu (8.8) jest 3-wymiarowa nad ciałem liczb rzeczywistych. Jej bazę stanowią pola  $\mathbf{g}_1$ ,  $\mathbf{g}_2$ ,  $[\mathbf{g}_1, \mathbf{g}_2]$ , a rząd wynosi 3. ■

### Przykład 8.2.2 (Układ Brocketta)

Wyznamy algebrę Liego układu (integratora) Brocketta o współrzędnych  $\mathbf{q} = (x, y, z)^\top$ , zadanego równaniami

$$\dot{\mathbf{q}} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ -y \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 1 \\ x \end{pmatrix} u_2 = \mathbf{g}_1(\mathbf{q})u_1 + \mathbf{g}_2(\mathbf{q})u_2.$$

<sup>S</sup>Zwane też *warunkiem rzędu* algebry Liego.

Wyliczając nawiasy Liego dla tego układu

$$[\mathbf{g}_1, \mathbf{g}_2] = \begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix}, \quad [\mathbf{g}_1, [\mathbf{g}_1, \mathbf{g}_2]] = [\mathbf{g}_2, [\mathbf{g}_1, \mathbf{g}_2]] = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix},$$

zauważamy, że jedynym nietrywialnym nawiasem jest  $[\mathbf{g}_1, \mathbf{g}_2]$ , a wszystkie nawiasy Liego składające się z co najmniej trzech generatorów znikają. Taką algebrę Liego<sup>¶</sup> nazywamy nilpotentną. Każda nilpotentna algebra Liego jest skończenie wymiarowa. ■

Przedstawimy teraz kilka własności nawiasu Liego, przydatnych do wyznaczania kierunków ruchu układów nieholonomicznych. Własności te można wyprowadzić bezpośrednio z definicji (8.7). Niech  $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$  oznaczają pola wektorowe, a  $\alpha, \beta \in \mathbb{R}$  będą liczbami rzeczywistymi. Wówczas zachodzą następujące własności:

- Dwuliniowość —  $[\alpha\mathbf{X} + \beta\mathbf{Y}, \mathbf{Z}] = \alpha[\mathbf{X}, \mathbf{Z}] + \beta[\mathbf{Y}, \mathbf{Z}]$ ,
- Skośna symetria —  $[\mathbf{X}, \mathbf{Y}] = -[\mathbf{Y}, \mathbf{X}]$ ,
- Tożsamość Jacobiego —  $[\mathbf{X}, [\mathbf{Y}, \mathbf{Z}]] + [\mathbf{Y}, [\mathbf{Z}, \mathbf{X}]] + [\mathbf{Z}, [\mathbf{X}, \mathbf{Y}]] = 0$ .

Jak wynika z przykładów 8.2.1, 8.2.2, algebra Liego układu może być konstruowana rekurencyjnie, rozpoczynając od generatorów. W naturalny sposób definiuje się *stopień* nawiasu Liego oddający „skomplikowanie” procedury generacyjnej przez określenie liczby generatorów w zapisie nawiasu Liego

$$\begin{cases} \text{stopień}(\mathbf{X}) = 1, & \text{dla generatorów,} \\ \text{stopień}([\mathbf{A}, \mathbf{B}]) = \text{stopień}(\mathbf{A}) + \text{stopień}(\mathbf{B}), & \text{dla pól złożonych.} \end{cases} \quad (8.9)$$

Dla przykładowego układu o dwóch generatorach  $\mathbf{X}, \mathbf{Y}$  stopnie wybranych pól wektorowych są następujące:  $\text{stopień}([\mathbf{X}, \mathbf{Y}], [[\mathbf{Y}, \mathbf{X}], \mathbf{Y}]) = 5$ ,  $\text{stopień}(\mathbf{Y}) = 1$ ,  $\text{stopień}([\mathbf{Y}, \mathbf{X}]) = 2$ . W zależności od stopnia, pola wektorowe w algebrze Liego można pogrupować w *warstwy*. Do warstwy  $i$ -tej zaliczamy pola stopnia  $i$ -tego

$$\mathbf{W}_i = \{\mathbf{X} \mid \text{stopień}(\mathbf{X}) = i\}. \quad (8.10)$$

<sup>¶</sup>A także układ sterowania z taką algebrą.



Szczególną klasą układów<sup>||</sup> są układy, których nawiasy Liego poczynając od pewnej warstwy zerują się. Są to *układy nilpotentne*, z nilpotentną algebrą Liego. Powiemy, że rząd układu nilpotentnego jest równy  $p$ , jeżeli

$$\mathbf{W}_p \neq \mathbf{W}_{p+1} = \{\mathbf{0}\}. \quad (8.11)$$

Przykładem układu nilpotentnego rzędu 2 jest układ Brocketta.

Do zbadania warunku rzędu algebry Liego wystarczy sprawdzić warunek rzędu dla sumy warstw  $\cup_{i=1}^{\infty} \mathbf{W}_i(\mathbf{q})$ . Okazuje się, że sprawdzenie tego warunku można uprościć przez wykorzystanie zależności między polami wektorowymi wynikającymi z własności skośnej symetrii i tożsamości Jacobiego. Nawiasy Liego pól wektorowych z algebry Liego, które są niezależne i rozpinają tę algebrę nad ciałem liczb rzeczywistych  $\mathbb{R}$  tworzą *bazę* algebry Liego. Jedna z konstrukcji bazy pochodzi od Ph. Halla i polega na wyborze pól wektorowych zgodnie z następującymi regułami:

**PH1** generatory należą do bazy,  $\mathbf{X}_i \in \mathcal{H}$ ,  $i = 1, \dots, m$ ,

**PH2** jeżeli  $\text{stopień}(\mathbf{B}_i) < \text{stopień}(\mathbf{B}_j)$ , wtedy  $\mathbf{B}_i \stackrel{\mathcal{H}}{<} \mathbf{B}_j$ ,

**PH3**  $[\mathbf{B}_i, \mathbf{B}_j] \in \mathcal{H}$  wtedy i tylko wtedy, gdy

a)  $\mathbf{B}_i, \mathbf{B}_j \in \mathcal{H}$  i  $\mathbf{B}_i \stackrel{\mathcal{H}}{<} \mathbf{B}_j$ , oraz

b) albo  $\mathbf{B}_j = \mathbf{X}_k$  dla pewnego  $k$ , albo  $\mathbf{B}_j = [\mathbf{B}_l, \mathbf{B}_r]$  dla pewnych  $\mathbf{B}_l, \mathbf{B}_r \in \mathcal{H}$  i  $\mathbf{B}_l \stackrel{\mathcal{H}}{\leq} \mathbf{B}_i$ .

Zauważmy, że reguła **PH2** nie jest warunkiem selekcyjnym, lecz wprowadzającym pewne uporządkowanie między warstwami pól. Pole z warstwy o numerze wyższym nie może poprzedzać w bazie Halla pola z warstwy o numerze niższym. Naturalnie, relacja częściowego porządku  $\mathbf{B}_l \stackrel{\mathcal{H}}{\leq} \mathbf{B}_i$  zachodzi, gdy albo  $\mathbf{B}_l \stackrel{\mathcal{H}}{<} \mathbf{B}_i$ , albo  $\mathbf{B}_l = \mathbf{B}_i$ .

### Przykład 8.2.3 (Algebra Liego o dwóch generatorach)

Elementy bazy Halla algebry Liego o dwóch generatorach  $\mathbf{X}, \mathbf{Y}$ , do warstwy piątej włącznie, mają postać

$$\begin{aligned} & \mathbf{X}, \mathbf{Y}, [\mathbf{X}, \mathbf{Y}], [\mathbf{X}, [\mathbf{X}, \mathbf{Y}]], [\mathbf{Y}, [\mathbf{X}, \mathbf{Y}]], \\ & [\mathbf{X}, [\mathbf{X}, [\mathbf{X}, \mathbf{Y}]]], [\mathbf{Y}, [\mathbf{X}, [\mathbf{X}, \mathbf{Y}]]], [\mathbf{Y}, [\mathbf{Y}, [\mathbf{X}, \mathbf{Y}]]], \\ & [\mathbf{X}, [\mathbf{X}, [\mathbf{X}, [\mathbf{X}, \mathbf{Y}]]]], [\mathbf{Y}, [\mathbf{X}, [\mathbf{X}, [\mathbf{X}, \mathbf{Y}]]]], [\mathbf{Y}, [\mathbf{Y}, [\mathbf{X}, [\mathbf{X}, \mathbf{Y}]]]], \\ & [\mathbf{Y}, [\mathbf{Y}, [\mathbf{Y}, [\mathbf{X}, \mathbf{Y}]]]], [[\mathbf{X}, \mathbf{Y}], [\mathbf{X}, [\mathbf{X}, \mathbf{Y}]]], [[\mathbf{X}, \mathbf{Y}], [\mathbf{Y}, [\mathbf{X}, \mathbf{Y}]]]. \quad \blacksquare \end{aligned}$$

<sup>||</sup>Zobacz przykład 8.2.2.

**Przykład 8.2.4 (Algebra Liego o trzech generatorach)**

Elementy bazy Halla, do warstwy trzeciej włącznie, algebry Liego o trzech generatorach  $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$  są następujące:

$$\begin{aligned} &\mathbf{X}, \mathbf{Y}, \mathbf{Z}, [\mathbf{X}, \mathbf{Y}], [\mathbf{X}, \mathbf{Z}], [\mathbf{Y}, \mathbf{Z}], \\ &[\mathbf{X}, [\mathbf{X}, \mathbf{Y}]], [\mathbf{X}, [\mathbf{X}, \mathbf{Z}]], [\mathbf{Y}, [\mathbf{X}, \mathbf{Y}]], [\mathbf{Y}, [\mathbf{X}, \mathbf{Z}]], \\ &[\mathbf{Y}, [\mathbf{Y}, \mathbf{Z}]], [\mathbf{Z}, [\mathbf{X}, \mathbf{Y}]], [\mathbf{Z}, [\mathbf{X}, \mathbf{Z}]], [\mathbf{Z}, [\mathbf{Y}, \mathbf{Z}]]. \quad \blacksquare \end{aligned}$$

Załóżmy, hipotetycznie, że każde kolejne (aż do wymiaru przestrzeni stanu  $n$ ) pole wektorowe z bazy Halla jest liniowo niezależne od wszystkich poprzednich pól. W takiej sytuacji, do zapewnienia sterowalności układu o dwóch generatorach ( $m = 2$ ) i trójwymiarowej przestrzeni stanu są wymagane dwie warstwy pól wektorowych\*\*. Gdy  $m = 2$ , a  $n = 4$ , potrzebne jest jeszcze jedno pole z warstwy trzeciej. Przykład 8.2.4 pokazuje, że ograniczając się do pól warstwy drugiej można zapewnić sterowalność układu o sześciowymiarowej przestrzeni stanu trzema sterowaniami,  $m = 3$ . Rozsądne wydaje się zatem oczekiwanie, że sterowanie układami nieholonomicznymi jest najtrudniejsze, gdy  $m = 2$ , a wymiar przestrzeni stanu jest duży.

W procedurze generacji bazy Halla algebry Liego układu wyznaczyliśmy potencjalne kierunki ruchu układu, bez wskazania, jak te kierunki zrealizować przy pomocy sterowań. Jedną z metod określania sterowań polega na wykorzystaniu *formuły Campbella-Bakera-Hausdorffa-Dynkina* (formuły CBHD). Formuła ta, prawdziwa dla analitycznych pól wektorowych  $\mathbf{X}, \mathbf{Y}$  i małych czasów  $t$ , ma postać

$$\begin{aligned} \exp(t\mathbf{X}) \exp(t\mathbf{Y}) = \exp\left( t\mathbf{X} + t\mathbf{Y} + \frac{t^2}{2}[\mathbf{X}, \mathbf{Y}] + \frac{t^3}{12}[[\mathbf{X}, \mathbf{Y}], \mathbf{Y}] + \right. \\ \left. - \frac{t^3}{12}[[\mathbf{X}, \mathbf{Y}], \mathbf{X}] - \frac{t^4}{24}[\mathbf{X}, [\mathbf{Y}, [\mathbf{X}, \mathbf{Y}]]] + \dots \right). \quad (8.12) \end{aligned}$$

Złożenie strumieni po lewej stronie formuły (8.12) należy odczytywać od strony lewej ku prawej, tzn. najpierw działa strumień pola  $\mathbf{X}$ , a następnie strumień pola  $\mathbf{Y}$ . Z postaci wzoru (8.12) można wywnioskować, że rezultatem złożenia strumieni pól  $\mathbf{X}$  i  $\mathbf{Y}$  jest przemieszczenie stanu wynikające z działania każdego z tych strumieni z osobna (składnik  $t\mathbf{X} + t\mathbf{Y}$ ) oraz z reszty szeregu, której najbardziej znaczącym składnikiem, dla małych czasów  $t$ , jest pole wektorowe  $[\mathbf{X}, \mathbf{Y}]$ .

\*\*Zobacz przykład 8.2.3.

**Przykład 8.2.5 (Generowanie nawiasów Liego)**

Wygenerujemy przy pomocy sterowań pole wektorowe  $[\mathbf{X}, \mathbf{Y}]$ . Stosując formułę CBHD do złożenia czterech segmentów sterowań uzyskujemy

$$\begin{aligned} \exp(t\mathbf{X}) \exp(t\mathbf{Y}) \exp(-t\mathbf{X}) \exp(-t\mathbf{Y}) &= \exp\left(t\mathbf{X} + t\mathbf{Y} + \right. \\ &+ \left. \frac{t^2}{2}[\mathbf{X}, \mathbf{Y}] + \dots\right) \exp\left(-t\mathbf{X} - t\mathbf{Y} + \frac{t^2}{2}[\mathbf{X}, \mathbf{Y}] + \dots\right) = \\ &= \exp\left(t^2[\mathbf{X}, \mathbf{Y}] + \dots\right). \end{aligned} \quad (8.13)$$

Sterowania generujące pole  $[\mathbf{X}, \mathbf{Y}]$  w efektywnym czasie  $t^2$  są łatwe do użycia, gdy (8.13) zapiszemy jako

$$\exp(t\mathbf{X} \cdot 1) \exp(t\mathbf{Y} \cdot 1) \exp(t\mathbf{X} \cdot -1) \exp(t\mathbf{Y} \cdot -1),$$

co oznacza, że sterowanie oddziałujące na generator  $\mathbf{X}$  jest włączone z amplitudą +1 przez czas  $t$ , następnie, przez taki sam czas, włączone jest sterowanie generatorem  $\mathbf{Y}$  z amplitudą +1, itd. Zwykle żąda się, by wygenerowane pole wektorowe działało przez przedział czasu o długości  $t$ , zatem po normalizacji czasu otrzymamy

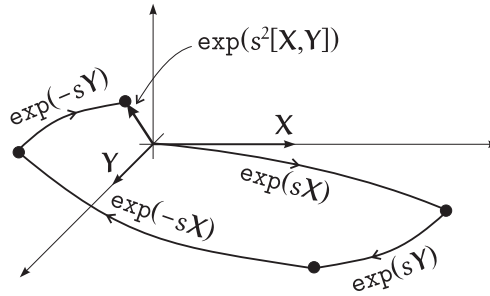
$$\begin{aligned} \exp(\sqrt{t}\mathbf{X}) \exp(\sqrt{t}\mathbf{Y}) \exp(-\sqrt{t}\mathbf{X}) \exp(-\sqrt{t}\mathbf{Y}) &= \\ = \exp\left(t[\mathbf{X}, \mathbf{Y}] + \frac{t^{3/2}}{2}([\mathbf{X}, [\mathbf{X}, \mathbf{Y}]] + [\mathbf{Y}, [\mathbf{X}, \mathbf{Y}]]) + \mathbf{o}(t^{3/2})\right). \end{aligned} \quad (8.14)$$

Generowanie pola drugiego stopnia  $[\mathbf{X}, \mathbf{Y}]$ , działającego przez krótki czas  $t$ , wymaga energii sterowań

$$\int_0^T \|\mathbf{u}\|^2 ds = \int_0^{\sqrt{t}} 1^2 ds + \int_0^{\sqrt{t}} (-1)^2 ds + \int_0^{\sqrt{t}} 1^2 ds + \int_0^{\sqrt{t}} (-1)^2 ds = 4\sqrt{t},$$

gdzie  $T$  jest całkowitym czasem sterowania. Zauważmy, że energia sterowań generujących pole  $\mathbf{X}$  lub pole  $\mathbf{Y}$  działające przez czas  $t$  wynosi tylko  $t$ . Spostrzeżenie to uogólnia się natychmiast na pola wyższych stopni. Ponadto, generowanie pól wyższych stopni wymaga większej liczby przełączeń sterowania. Stąd wniosek, że jeżeli dwa pola różniące się stopniem realizują cel sterowania, to należy preferować pole niższego stopnia jako bardziej efektywne energetycznie i zapewniające mniejszą liczbę przełączeń. ■

Geometryczną konstrukcję nawiasu Liego zgodną z analitycznym zapisem (8.14) zamieszczono na rysunku 8.6. Rysunek ten wskazuje, że aby uzyskać ruch w kierunku pola  $[\mathbf{X}, \mathbf{Y}]$ , należy wykonać dość obszerny ruch

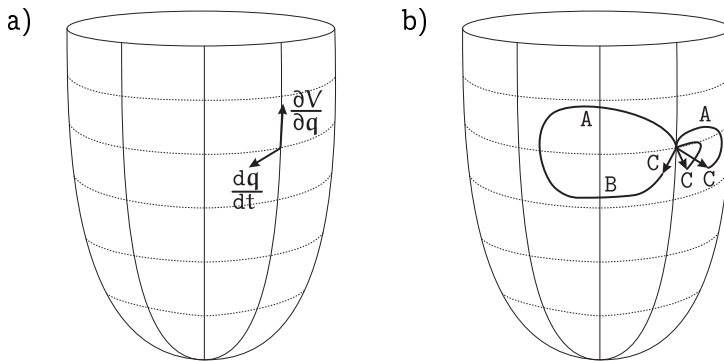


Rysunek 8.6 Geometryczna interpretacja nawiasu Liego,  $s^2 = t$ .

w kierunku pól-generatorów  $X$ ,  $Y$ . Stąd też, do wykazania zbieżności metod bazujących na ruchu w kierunku pól wektorowych nie wystarcza II metoda Lapunowa (zobacz rysunek 8.7a) żądająca monotonicznego zmniejszania odległości do stanu docelowego mierzonej funkcją Lapunowa  $V$  zgodnie z analitycznym warunkiem  $dV/dt = (\partial V/\partial \mathbf{q}) \cdot \dot{\mathbf{q}} < 0$ . Osłabienie wymagań II metody Lapunowa (zilustrowane na rysunku 8.7b) polega na wymaganiu, by trajektoria rozpoczynająca się w punkcie bieżącym przestrzeni stanu, znalazła się po czasie  $t$  w pewnym punkcie (C) bliższym stanu końcowego niż jej punkt początkowy, ale dopuszcza się, żeby w międzyczasie punkty trajektorii (A) oddalały się od stanu docelowego.

### 8.3 Komentarze i uwagi bibliograficzne

W literaturze można znaleźć liczne przykłady nieholonomicznych układów robotycznych z ograniczeniami fazowymi w postaci Pfaffa: kołowe roboty mobilne (także z przyczepami) [ANBC91, Hos96, TLM<sup>+</sup>92], roboty podwodne [EBS94], roboty szybujące [MN92, UY88], manipulator nieholonomiczny [SNC94], chwytaki nieholonomiczne [MLS94]. Przykład ograniczeń nieholonomicznych, które nie są w postaci Pfaffa, można znaleźć w pracy [LMO97]. Przykład nilpotentnego układu nieholonomicznego zwanego układem Brocketta pochodzi z pracy Brocketta [Bro81]. Spadający kot był przedmiotem zainteresowania fizjologa Mcdonalda [Mcd55]. Fałszywość powszechnego przeświadczenia o aktywnym udziale ogona w manewrowaniu spadającego kota wykazała praca [KS69]. Podstawowe wiadomości o grupach i algebrach Liego można znaleźć m.in. w [Gan87, Ser65, Var74, War71, Woj86]. Definicje podstawowych pojęć wykorzystywanych przy planowaniu ruchu zawiera artykuł [MS93]. Twierdzenie będące podstawą teore-



**Rysunek 8.7** a) Zbieżność według II metody Lapunowa — monotoniczne zmniejszanie odległości do celu. b) Zbieżność dla metod planowania ruchu układów nieholonomicznych — wypadkowy ruch (C) zmniejsza odległość do celu, lecz zwykle nie monotonicznie.

tyczną wielu metod planowania ruchu robotów nieholonomicznych sformułował Chow [Cho39]. Algorytm konstrukcji bazy Halla algebry Liego układu można znaleźć w artykule [MS93], natomiast obliczeniowo optymalny sposób generacji tej bazy przedstawiono w pracy [Dul97a]. Formułę Campbella-Bakera-Hausdorffa-Dynkina służącą do generacji kierunków ruchu przy pomocy sterowań odcinkami stałych zaczerpnięto z prac Hermesa [Her78, Her80], można ją także znaleźć w każdym podręczniku dotyczącym algebr Liego. Rozważania na temat złożoności zadania planowania ruchu robotów nieholonomicznych tylko na podstawie analizy wymiaru przestrzeni stanu i wymiaru przestrzeni sterowań zawiera praca [Dul97b]. W myśl tej pracy, za trudne uważa się takie zadania planowania ruchu, dla których do spełnienia warunku rzędu algebry Liego jest wymagane wygenerowanie wielu warstw pól wektorowych w algebrze Liego układu. Mnemotechniczna reguła określania stopnia trudności zadania planowania ruchu stanowi, że zadanie jest tym trudniejsze, im większy jest wymiar przestrzeni stanu, a mniejsza liczba sterowań.

## Literatura

- [ANBC91] B. d'Andréa-Novel, G. Bastin i G. Campion, Modelling and control of non holonomic wheeled mobile robots. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 1130–1135, Sacramento, 1991.

- [Bro81] R. W. Brockett, Control theory and singular Riemannian geometry. W: P. Hinton i G. Young, (red.), *New Directions in Applied Mathematics*, strony 11–27. Springer-Verlag, New York, 1981.
- [Cho39] W. L. Chow, Über Systeme von linearen partiellen Differentialgleichungen erster Ordnung. *Math. Ann.*, **117**(1):98–105, 1939.
- [Dul97a] I. Dułęba, Checking controllability of nonholonomic systems via optimal Ph. Hall basis generation. W: *Proc. IFAC SyRoCo Conference*, vol. 2, strony 485–490, Nantes, 1997.
- [Dul97b] I. Dułęba, Teoretyczne aspekty planowania ruchu kołowych robotów mobilnych. W: *Warsztaty Robotów Mobilnych*, strony 111–118, Zakopane, 1997.
- [EBS94] O. Egeland, E. Berglund i O. J. Sjørdalen, Exponential stabilization of a nonholonomic underwater vehicle with constant desired configuration. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 20–25, San Diego, 1994.
- [Gan87] J. Gancarzewicz, *Geometria różniczkowa*. PWN, Warszawa, 1987.
- [Her78] H. Hermes, Lie algebras of vector fields and local approximation of attainable sets. *SIAM J. Contr. Opt.*, **16**(6):715–728, 1978.
- [Her80] H. Hermes, On the synthesis of a stabilizing feedback control via Lie algebraic method. *SIAM J. Contr. Opt.*, **18**(6):352–361, 1980.
- [Hos96] R. Hossa, *Modele i algorytmy sterowania kołowych robotów mobilnych*. Rozprawa doktorska, Instytut Cybernetyki Technicznej, Politechnika Wrocławska, 1996.
- [KS69] T.R. Kane i M.P. Scher, A dynamical explanation of the falling cat phenomenon. *Int. J. Solid Structures*, strony 663–670, 1969.
- [LMO97] A. De Luca, R. Mattone i G. Oriolo, Control of redundant robots under end-effector commands: A case study in underactuated systems. *Appl. Mathematics and Computer Science*, **7**(2):225–251, 1997.
- [Mcd55] D.A. Mcdonald, How does a falling cat turn over. *J. Physiol. Paris*, **129**:34–35, 1955.
- [MLS94] R. M. Murray, Z. Li i S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton, 1994.
- [MN92] R. Mukherjee i Y. Nakamura, Formulation and efficient computation of inverse dynamics of space robots. *IEEE Trans. Robotics Automat.*, **3**(8):400–406, 1992.

- [MS93] R. M. Murray i S. Sastry, Nonholonomic motion planning: Steering using sinusoids. *IEEE Trans. Autom. Contr.*, **38**(5):700–716, 1993.
- [Ser65] J. P. Serre, *Lie Algebras and Lie groups*. W.J. Benjamin, New York, 1965.
- [SNC94] O. J. Sørдалen, Y. Nakamura i W. J. Chung, Design of a nonholonomic manipulator. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. **1**, strony 8–13, San Diego, 1994.
- [TLM<sup>+</sup>92] D. Tilbury, J. P. Laumond, R. Murray, S. Sastry i G. Walsh, Steering car-like systems with trailers using sinusoids. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. **2**, strony 1993–1998, Nice, 1992.
- [UY88] Y. Umetani i K. Yoshida, Theoretical and experimental study on in-orbit capture operation with satellite mounted manipulator. W: *Proc. IFAC Symp. Autom. Contr. Aerospace*, strony 137–142, 1988.
- [Var74] V. S. Varadarayan, *Lie groups, Lie algebras and their representations*. Prentice-Hall Inc., 1974.
- [War71] F. W. Warner, *Foundations of Differentiable Manifolds and Lie Groups*. Scott-Foresmen, Glenview, 1971.
- [Woj86] W. Wojtyński, *Grupy i algebry Liego*. PWN, Warszawa, 1986.





## Rozdział 9

# Metody planowania ruchu ogólnego przeznaczenia

Niniejszy rozdział zawiera przegląd metod planowania ruchu robota mobilnego opisanego równaniami kinematyki (2.145). Cechą charakterystyczną prezentowanych metod jest przyjęcie bardzo słabych założeń w odniesieniu do kinematyki robota mobilnego. Oprócz spełnienia warunku sterowalności układu (w krótkim czasie), nie jest wymagane spełnienie żadnego innego warunku zarówno dotyczącego struktury równań układu, jak i jego szczególnych własności. W kolejnych podrozdziałach zostaną omówione:

- Metoda globalna planowania ruchu wykorzystująca Zasadę Maksimum Pontriagina. Charakteryzuje się ona dużą złożonością obliczeniową, lecz pozwala na optymalizację globalnych kryteriów jakości wynikowej trajektorii ruchu.
- Metoda Newtona wprowadzona w podrozdziale 4.1. Metoda ta wymaga prowadzenia czasochłonnej iteracyjnej procedury obliczeniowej, a przy tym nie optymalizuje żadnego kryterium jakości trajektorii.
- Globalna metoda planowania ruchu bazująca na technice uśredniania. Postulatem konstytuującym tę metodę jest założenie o możliwości dekompozycji, z małym błędem, kinematyki robota mobilnego (2.145) na dwa podukłady sterowane odpowiednio przebiegami wolno- i szybkozmiennymi. Metoda uśredniania nie optymalizuje żadnego kryterium jakości trajektorii.
- Lokalna metoda Lie-algebraiczna planowania ruchu. Metoda najmniej złożona obliczeniowo i pozwalająca na optymalizację lokalnych kryteriów jakości trajektorii.

## 9.1 Metoda oparta na Zasadzie Maksimum Pontriagina

Ogólnej metody rozwiązania zadań sterowania optymalnego dostarcza *Zasada Maksimum Pontriagina*. Dla ustalenia uwagi, niech zadanie planowania ruchu polega na przemieszczeniu stanu układu nieholonomicznego

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u}$$

od punktu  $\mathbf{q}_0$  do punktu  $\mathbf{q}_d$ , w zadanym przedziale czasu  $[0, T]$ , w taki sposób by optymalizować energetyczną funkcję jakości

$$\mathcal{J}(\mathbf{u}(\cdot)) = \frac{1}{2} \int_0^T \langle \mathbf{u}(t), \mathbf{C}\mathbf{u}(t) \rangle dt,$$

gdzie  $\mathbf{C}$  jest symetryczną i dodatnio określoną stałą macierzą wagową. Sterowania nie podlegają ograniczeniom. W takim przypadku hamiltonian ma postać

$$H(\mathbf{q}, \mathbf{p}, \mathbf{u}) = -\frac{1}{2} \langle \mathbf{u}, \mathbf{C}\mathbf{u} \rangle + \langle \mathbf{p}, \mathbf{G}(\mathbf{q})\mathbf{u} \rangle, \quad (9.1)$$

gdzie  $\mathbf{p} \in \mathbb{R}^n$  oznacza zmienną dołączoną. Zasada Maksimum Pontriagina stanowi\*, że warunkiem koniecznym optymalności sterowania  $\mathbf{u}^*$  jest

$$H(\mathbf{q}, \mathbf{p}, \mathbf{u}^*) = \max_{\mathbf{u}} H(\mathbf{q}, \mathbf{p}, \mathbf{u}) = \text{const}, \quad (9.2)$$

przy spełnieniu równań kanonicznych Hamiltona

$$\begin{cases} \dot{\mathbf{q}} = \frac{\partial H(\mathbf{q}, \mathbf{p}, \mathbf{u})}{\partial \mathbf{p}} = \mathbf{G}(\mathbf{q})\mathbf{u} \\ \dot{\mathbf{p}} = -\frac{\partial H(\mathbf{q}, \mathbf{p}, \mathbf{u})}{\partial \mathbf{q}} = -\left(\frac{\partial(\mathbf{G}(\mathbf{q})\mathbf{u})}{\partial \mathbf{q}}\right)^T \mathbf{p}. \end{cases} \quad (9.3)$$

Trójkę  $(\mathbf{u}(t), \mathbf{q}(t), \mathbf{p}(t))$  spełniającą równania (9.2) i równania kanoniczne Hamiltona nazywamy *ekstremalą*. Z warunku (9.2), ekstremum względem sterowań uzyskuje się, gdy

$$\frac{\partial H(\mathbf{q}, \mathbf{p}, \mathbf{u})}{\partial \mathbf{u}} = \mathbf{0}. \quad (9.4)$$

---

\*Zobacz dodatek A.5.

Sterowania wyznaczone z (9.4), w zależności od stanu i zmiennej dołączonej są następujące

$$-\mathbf{C}\mathbf{u} + \mathbf{G}^T(\mathbf{q})\mathbf{p} = 0 \quad \Rightarrow \quad \mathbf{u} = \mathbf{C}^{-1}\mathbf{G}^T(\mathbf{q})\mathbf{p}.$$

Podstawiając wyznaczone sterowania do równań kanonicznych Hamiltona uzyskujemy zadanie dwubrzegowe

$$\begin{cases} \dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{C}^{-1}\mathbf{G}^T(\mathbf{q})\mathbf{p} \\ \dot{\mathbf{p}} = -\left(\frac{\partial}{\partial \mathbf{q}}\mathbf{G}(\mathbf{q})\mathbf{C}^{-1}\mathbf{G}^T(\mathbf{q})\mathbf{p}\right)^T \mathbf{p}, \end{cases} \quad (9.5)$$

z zadanymi  $T$ ,  $\mathbf{q}(0) = \mathbf{q}_0$  i  $\mathbf{q}(T) = \mathbf{q}_d$ . Zadanie dwubrzegowe jest skomplikowanym zadaniem obliczeniowym wymagającym zastosowania specjalnych metod rozwiązania, np. metody strzałów.

Zaletami metody pontriaginowskiej planowania ruchu jest jej ogólność i możliwość optymalizacji globalnej ruchu. Do wad metody należą:

1. Konieczność wykazania istnienia rozwiązania optymalnego, a zatem zasadności zastosowania Zasady Maksimum Pontriagina.
2. Trudności w rozwiązaniu numerycznym zadania dwubrzegowego.
3. Gwarantowana zbieżność tylko wtedy, gdy trajektoria początkowa dla zadania dwubrzegowego jest bliska trajektorii optymalnej — w przeciwnym przypadku zbieżność może być trudna do uzyskania.
4. Duży wymiar zadania (minimalnie  $2n$ , gdzie  $n$  jest liczbą zmiennych stanu) — dla najprostszych robotów mobilnych  $n = 3$ , gdy zadanie jest rozważane na poziomie kinematycznym; po uwzględnieniu dynamiki robota, wymiar zadania rośnie dwukrotnie.
5. Ograniczenia nałożone na przestrzeń stanu (np. z powodu istnienia przeszkód) znacznie ograniczające efektywność zastosowania Zasady Maksimum Pontriagina.

## 9.2 Metoda Newtona

Podstawowy schemat metody Newtona został przedstawiony w podrozdziale 4.1, w kontekście odwrotnego zadania kinematyki układu nieholonomicznego (robota mobilnego). W niniejszym podrozdziale zwrócimy szczególną

uwagę na aspekty obliczeniowe metody Newtona. Przypomnijmy, że kinematyka nieholonomicznego robota mobilnego może być określona za pośrednictwem układu sterowania (2.145)

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u}, \quad (9.6)$$

lub odwzorowania (2.147)

$$\mathbf{k}_{\mathbf{q}_0, T} : \mathbb{L}_m^2[0, T] \longrightarrow \mathbb{R}^n, \quad \mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot)) = \boldsymbol{\varphi}_{T, \mathbf{q}_0}(\mathbf{u}(\cdot)), \quad (9.7)$$

gdzie  $\mathbf{q}_0$  jest stanem początkowym układu, a  $T$  przedziałem czasu sterowania. Mając zadany stan końcowy  $\mathbf{q}_d$  układu oraz jacobian analityczny  $D\mathbf{k}_{\mathbf{q}_0, T}$  układu nieholonomicznego, sterowanie pozwalające osiągnąć  $\mathbf{q}_d$  w chwili  $T$  obliczamy według algorytmu (4.8)

$$\frac{d\mathbf{u}_\tau(\cdot)}{d\tau} = -\alpha D\mathbf{k}_{\mathbf{q}_0, T}^\#(\mathbf{u}_\tau(\cdot))(\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}_\tau(\cdot)) - \mathbf{q}_d).$$

W wersji dyskretnej algorytm (4.8) ma postać

$$\mathbf{u}_{i+1}(\cdot) = \mathbf{u}_i(\cdot) - \xi_i D\mathbf{k}_{\mathbf{q}_0, T}^\#(\mathbf{u}_i(\cdot))(\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}_i(\cdot)) - \mathbf{q}_d). \quad (9.8)$$

Jeśli  $\mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}_i) - \mathbf{q}_d = 0$ , to stan docelowy  $\mathbf{q}_d$  został osiągnięty i zadanie planowania jest rozwiązane. W przeciwnym przypadku sterowanie  $\mathbf{u}_i$  jest modyfikowane zgodnie ze wzorem (9.8). Parametr  $\xi_i$  jest małą, dodatnią liczbą wpływającą na szybkość zbieżności metody Newtona. Dobór  $\xi_i$  wymaga dużej staranności, albowiem przyjęcie zbyt dużych wartości  $\xi_i$  powoduje chaotyczne zmiany generowanych trajektorii, natomiast zbyt małych — wolną zbieżność procesu iteracyjnego. Dobrze umotywowanym teoretycznie wyborem zmian  $\xi_i$  jest zapewnienie spełnienia dwóch warunków

$$\sum_{i=1}^{\infty} \xi_i = \infty, \quad \sum_{i=1}^{\infty} \xi_i^2 < \infty, \quad (9.9)$$

co zachodzi np. dla  $\xi_i = 1/i$ . Pierwszy z warunków (9.9) zapewnia penetrację całej przestrzeni sterowań, drugi opisuje skończoną „energię” współczynników  $\xi_i$ , co w szczególności oznacza  $\xi_i \rightarrow 0$  przy  $i \rightarrow +\infty$ .

Zastosowanie formuły (9.8) wymaga podziału osi czasu na  $N$  przedziałów\* oraz przyjęciu na każdym z przedziałów stałych sterowań. Sterowania odcinkami stałe są modyfikowane według (9.8). Liczba modyfikowanych

\* $N$  musi być odpowiednio duże, by podział nie wprowadzał istotnych błędów.

sterowań jest bardzo duża i wynosi  $m \cdot N$ , ponadto sterowania odcinkami stałe są w praktyce trudne do realizacji. Dlatego przyjmuje się sterowania w postaci pewnego szeregu funkcyjnego. Niech szereg ten będzie zadany formułą

$$\mathbf{u}(t, \boldsymbol{\lambda}) = \sum_{i=1}^{\infty} \lambda_i \phi_i(t), \quad (9.10)$$

dla  $t \in [0, T]$ , gdzie funkcje  $\phi_i(t)$  tworzą bazę przestrzeni funkcyjnej  $L_m^2[0, T]$ , (np. bazę Fouriera), natomiast  $\lambda_i$  są wektorami współczynników rozwinięcia (9.10) należącymi do przestrzeni  $l_m^2$ . W tym przypadku algorytm Newtona (9.8) modyfikuje sterowania przez dostrajanie parametrów  $\lambda_i$ , bez zmiany elementów bazy. W implementacji komputerowej, sterowania przyjmuje się w postaci sumy skończonej

$$u_k(t) = \sum_{i=1}^{N^k} \lambda_i^k \phi_i(t), \quad k = 1, \dots, m, \quad t \in [0, T], \quad (9.11)$$

gdzie  $N^k$  jest liczbą elementów szeregu dla  $k$ -tego sterowania. Wykorzystując rozważania z podrozdziału 4.1 znajdziemy odpowiednik formuły (9.8) w przestrzeni współczynników  $\boldsymbol{\lambda}$ .

Odwzorowanie opisujące punkt w przestrzeni stanu osiągnięty w chwili  $T$  z punktu początkowego  $\mathbf{q}_0$  pod działaniem sterowania (9.11) jest zdefiniowane przez

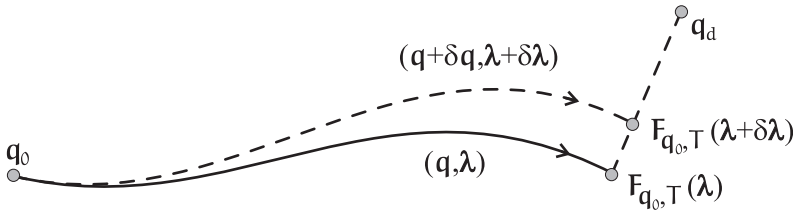
$$\mathbf{F}_{\mathbf{q}_0, T}(\boldsymbol{\lambda}) = \mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot, \boldsymbol{\lambda})), \quad (9.12)$$

gdzie wektor parametrów  $\boldsymbol{\lambda} = (\lambda_1^1, \dots, \lambda_{N^1}^1, \lambda_1^2, \dots, \lambda_{N^2}^2, \dots, \lambda_1^m, \dots, \lambda_{N^m}^m)^T$  składa się z  $N = \sum_{i=1}^m N^i$  elementów. Ponieważ zakładamy, że odwzorowanie  $\mathbf{F}_{\mathbf{q}_0, T}(\boldsymbol{\lambda}) \neq \mathbf{q}_d$ , konieczna jest zatem modyfikacja wektora parametrów  $\delta\boldsymbol{\lambda}$ . Wymagamy, by zmiana parametrów o  $\delta\boldsymbol{\lambda}$  spowodowała przemieszczenie stanu  $\mathbf{F}_{\mathbf{q}_0, T}(\boldsymbol{\lambda} + \delta\boldsymbol{\lambda})$  w kierunku zadanego stanu docelowego

$$\mathbf{F}_{\mathbf{q}_0, T}(\boldsymbol{\lambda} + \delta\boldsymbol{\lambda}) - \mathbf{F}_{\mathbf{q}_0, T}(\boldsymbol{\lambda}) = \frac{\partial \mathbf{F}_{\mathbf{q}_0, T}(\boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}} \delta\boldsymbol{\lambda} = \xi(\mathbf{q}_d - \mathbf{F}_{\mathbf{q}_0, T}(\boldsymbol{\lambda})),$$

co zobrazowano na rysunku 9.1. By wyliczyć  $\frac{\partial \mathbf{F}_{\mathbf{q}_0, T}(\boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}} \delta\boldsymbol{\lambda}$  wykorzystamy zależność

$$\frac{\partial \mathbf{F}_{\mathbf{q}_0, T}(\boldsymbol{\lambda})}{\partial \boldsymbol{\lambda}} \delta\boldsymbol{\lambda} = D \mathbf{k}_{\mathbf{q}_0, T}(\mathbf{u}(\cdot, \boldsymbol{\lambda})) \delta\mathbf{u}(\cdot, \boldsymbol{\lambda}). \quad (9.13)$$



**Rysunek 9.1** Zasada konstrukcji zaburzenia wektora  $\delta\lambda$  w parametrycznym algorytmie Newtona.

Jakobian analityczny kinematyki robota mobilnego (2.151) ma postać

$$\begin{aligned} D \mathbf{k}_{q_0, T}(\mathbf{u}(\cdot, \lambda)) \delta \mathbf{u}(\cdot, \lambda) &= \int_0^T \mathbf{\Phi}_\lambda(T, s) \mathbf{G}_\lambda(s) \frac{\partial \mathbf{u}(s, \lambda)}{\partial \lambda} ds \delta \lambda = \\ &= \int_0^T \mathbf{\Phi}_\lambda(T, s) \mathbf{B}_\lambda(s) ds \delta \lambda. \end{aligned} \quad (9.14)$$

Gdy  $\mathbf{G} = [\mathbf{g}_1 \mathbf{g}_2 \cdots \mathbf{g}_m]$ , wtedy macierz  $\mathbf{B}_\lambda(t)$  wymiaru  $n \times N$  wyliczana jest jak następuje

$$\mathbf{B}_\lambda(t) = [\mathbf{g}_1 \boldsymbol{\phi}^1 \mathbf{g}_2 \boldsymbol{\phi}^2 \cdots \mathbf{g}_m \boldsymbol{\phi}^m],$$

gdzie  $\boldsymbol{\phi}^k = (\phi_1, \dots, \phi_{N^k})^T$  oraz  $\mathbf{g}_k \boldsymbol{\phi}^k = [\mathbf{g}_k \phi_1 \mathbf{g}_k \phi_2 \cdots \mathbf{g}_k \phi_{N^k}]$ . Występująca w równaniu (9.14) macierz  $\mathbf{\Phi}_\lambda(t, s)$  jest rozwiązaniem równania

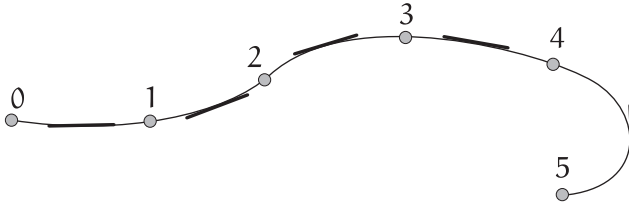
$$\frac{d}{dt} \mathbf{\Phi}_\lambda(t, s) = \mathbf{A}_\lambda(t) \mathbf{\Phi}_\lambda(t, s), \quad (9.15)$$

w którym  $\mathbf{A}_\lambda(t) = \frac{\partial (\mathbf{G}(\mathbf{q}(t)) \mathbf{u}(t, \lambda))}{\partial \mathbf{q}}$  przy warunku początkowym  $\mathbf{\Phi}_\lambda(s, s) = \mathbb{I}_n$ .

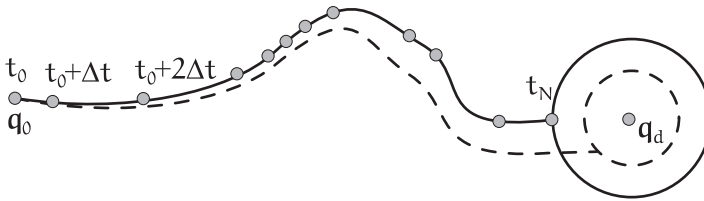
Procedura wyznaczania  $\mathbf{\Phi}_\lambda(T, s)$  jest następująca. Na początku przedział  $[s, T]$  jest dzielony na  $r$  podprzedziałów o długości  $\Delta s$ . Następnie, w każdym z podprzedziałów wybiera się punkt  $\tau_k$ , w którym jest wyliczana macierz  $\mathbf{A}_\lambda(\tau_k)$ . Ostatecznie,  $\mathbf{\Phi}_\lambda(T, s)$  jest aproksymowane zależnością

$$\begin{aligned} \mathbf{\Phi}_\lambda(T, s) &= \lim_{\Delta s \rightarrow 0} (\mathbb{I}_n + \mathbf{A}_\lambda(\tau_r) \Delta s) (\mathbb{I}_n + \mathbf{A}_\lambda(\tau_{r-1}) \Delta s) \cdots \\ &\quad \cdots (\mathbb{I}_n + \mathbf{A}_\lambda(\tau_1) \Delta s) \end{aligned} \quad (9.16)$$

z  $\Delta s = \frac{T-s}{r}$ . Aproksymacyjny charakter formuły (9.16) został zilustrowany na rysunku 9.2. Zauważmy, że równomierny podział przedziału czasu niekoniecznie musi powodować równomierny podział trajektorii (zobacz rysunek 9.3), ponadto dynamika zmian  $\mathbf{A}_\lambda(\tau)$  również zależy od dyskretyzacji.



**Rysunek 9.2** Wpływ dyskretyzacji na dokładność schematu iteracyjnego (9.16). Jednorodny podział przedziału  $[0, s]$  prowadzi do błędów aproksymacji (między punktami 4 i 5), natomiast zmienna dyskretyzacja przedziału  $[0, s]$  utrudnia implementację algorytmu.



**Rysunek 9.3** Zasada dowodu zbieżności metody Newtona — zmiana parametru powoduje zmniejszenie odległości euklidesowej do stanu końcowego.

Łącząc równania (9.13), (9.14) i uwzględniając, że  $\delta\lambda$  nie zależy od czasu  $s$ , mamy

$$\int_0^T \Phi_{\lambda}(T, s) \mathbf{B}_{\lambda}(s) ds \delta\lambda = D \mathbf{k}_{q_0, T}(\mathbf{u}(\cdot, \lambda)) \delta \mathbf{u}(\cdot, \lambda) = -\xi(\mathbf{F}_{q_0, T}(\lambda) - \mathbf{q}_d). \quad (9.17)$$

Z równania (9.17) wyznaczamy  $\delta\lambda$  według zależności

$$\lambda_{i+1} = \lambda_i - \xi_i D \mathbf{k}_{q_0, T}^{\#}(\mathbf{u}(\cdot, \lambda_i)) (\mathbf{F}_{q_0, T}(\lambda_i) - \mathbf{q}_d). \quad (9.18)$$

Zaletą metody Newtona w wersji parametrycznej jest gwarantowana ciągłość wynikowych sterowań, ograniczeniem — występowanie w procedurze obliczeniowej skończonego rozwinięcia szeregu (9.11) (górną granicę sumowania wynosi  $M = \max_k (N^k) < \infty$ ). Przyjęcie zbyt małej wartości  $M$  powoduje utratę zbieżności schematu iteracyjnego Newtona, przyjęcie zbyt dużej — wzrost złożoności obliczeniowej. Sterowania  $\mathbf{u}_i$ , przy właściwej dyskretyzacji, wyliczone zgodnie ze wzorem (9.11) dla wektora  $\lambda_i$  danego zależnością (9.18) zapewniają zmniejszanie, w kolejnych iteracjach, odległości od stanu docelowego  $\mathbf{q}_d$  (zobacz rysunek 9.3), a zatem  $\mathbf{F}_{q_0, T}(\lambda_i) \rightarrow \mathbf{q}_d$ .

Na zakończenie, podsumujemy zalety i wady metody Newtona. Wśród zalet zauważamy dobre uzasadnienie teoretyczne metody i szeroki zakres jej stosowalności. Wadami metody są: duża złożoność obliczeniowa, konieczność (dynamicznego) doboru parametrów warunkujących zbieżność algorytmu (liczba i rozkład punktów dyskretyzujących zlinearyzowane równanie układu) oraz możliwość złego uwarunkowania numerycznego, np. przy występowaniu osobliwości odwracanych macierzy.

### 9.3 Metoda uśredniania

Zadania planowania ruchu można także rozwiązywać znaną od ponad stu lat w matematyce stosowaną *metodą uśredniania*. Metoda ta jest wykorzystywana przy założeniu, że ruch układu sterowanego może być zdekomponowany na składową wolnozmienną i szybkozmienną oscylację (zaburzenia) wokół trajektorii odpowiadającej przebiegowi wolnozmiennemu. Najpierw przebiegi wolno- i szybkozmienną opisuje się przy pomocy kombinacji liniowej elementów pewnej bazy, a następnie wylicza nieznaną współczynniki tej kombinacji. Ideę metody uśredniania zilustrowano graficznie na rysunku 9.4. Poniżej zostanie przedstawione podstawowe twierdzenie uzasadniające stosowalność tej metody.

Niech układ nieholonomiczny

$$\dot{\mathbf{q}} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{q}) u_i^\varepsilon \quad (9.19)$$

będzie sterowany sumą dwóch przebiegów

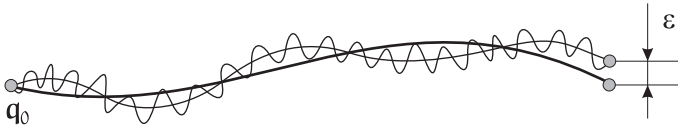
$$u_i^\varepsilon(t) = u_i^s(t) + \frac{1}{\sqrt{\varepsilon}} u_i^f(t, \vartheta), \quad (9.20)$$

gdzie  $\vartheta = t/\varepsilon$ . Funkcja  $u_i^f$  jest okresowa ze względu na argument  $\vartheta$  z okresem  $2\pi$  i ma zerową wartość średnią

$$\int_0^{2\pi} u_i^f(t, \vartheta) d\vartheta = 0.$$

Szybkozmienną składową sterowania opisuje  $\vartheta$ , natomiast  $t$ ,  $\mathbf{q}$  są przebiegami wolnozmiennymi. Stosując twierdzenie o uśrednianiu możemy dla odpowiednio małego parametru  $\varepsilon$  przybliżyć rozwiązanie układu (9.19) rozwiązaniem układu uśrednionego nie zawierającego zmiennej  $\vartheta$ .





**Rysunek 9.4** Idea metody uśredniania: możliwe jest przybliżenie, z dowolnie małym błędem, trajektorii poszukiwanej sumą dwóch trajektorii — wolnozmiennnej i szybkozmiennnej.

**Twierdzenie 9.3.1** Dla odpowiednio małej wartości parametru  $\varepsilon$  trajektoria układu (9.19) jest przybliżoną trajektorią układu

$$\dot{\mathbf{z}} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{z}) u_i^s(t) + \frac{1}{2\pi} \sum_{\substack{i,j=1 \\ i < j}}^m [\mathbf{g}_i, \mathbf{g}_j](\mathbf{z}) \eta_{ij}(t),$$

$\mathbf{z}(0) = \mathbf{q}(0) = \mathbf{q}_0$ , w sensie metryki Czebyszewa

$$\max_{t \in [0, 2\pi]} |\mathbf{z}(t) - \mathbf{q}(t)| = \|\mathbf{z} - \mathbf{q}\|_{C[0, 2\pi]} \leq \Delta_\varepsilon,$$

gdzie parametr  $\Delta_\varepsilon$  dąży do zera, gdy  $\varepsilon \rightarrow 0$ . Funkcja  $\eta_{ij}$  zależy od sterowań szybkozmiennych

$$\eta_{ij} = \int_0^{2\pi} \int_0^\vartheta u_i^f(t, \tau) u_j^f(t, \vartheta) d\tau d\vartheta. \quad (9.21)$$

Prześledźmy teraz, w jaki sposób metoda uśredniania jest stosowana w sterowaniu układów nieholonomicznych. Rozważmy kinematykę układu nieholonomicznego

$$\dot{\mathbf{q}} = \mathbf{g}_1(\mathbf{q}) u_1 + \mathbf{g}_2(\mathbf{q}) u_2 + \dots + \mathbf{g}_m(\mathbf{q}) u_m \quad (9.22)$$

ze stanem początkowym  $\mathbf{q}_0$  i końcowym  $\mathbf{q}_d$ . Załóżmy, że generatory wraz z nawiasami Liego drugiego stopnia rozpinają w każdym punkcie przestrzeni stanu  $\mathbb{R}^n$ , tj. macierz  $\mathbf{C} = [\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_m, [\mathbf{g}_1, \mathbf{g}_2], [\mathbf{g}_1, \mathbf{g}_3], \dots, [\mathbf{g}_{m-1}, \mathbf{g}_m]]$  jest wszędzie pełnego rzędu. Algorytm planowania ruchu wykorzystujący metodę uśredniania składa się z następujących kroków:

**Krok 1.** Wybrany holonomiczny układ planowania ruchu wyznacza trajektorię gładką  $\mathbf{y}(t)$ , łączącą punkty  $\mathbf{q}_0$  z  $\mathbf{q}_d$ . Przez różniczkowanie tej trajektorii uzyskujemy równanie  $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$  z warunkiem początkowym  $\mathbf{y}(0) = \mathbf{q}_0$ , które spełnia trajektoria.

**Krok 2.** Wyliczamy sterowanie rozszerzone  $\mathbf{v} = (v_1, \dots, v_m, v_{m+1}, \dots, v_{m+l})^T$ ,  $l = \frac{m(m-1)}{2} \geq n$  z równania liniowego

$$[\mathbf{g}_1, \dots, \mathbf{g}_m, [\mathbf{g}_1, \mathbf{g}_2], \dots, [\mathbf{g}_{m-1}, \mathbf{g}_m]] \mathbf{v} = \mathbf{C}(\mathbf{y}) \mathbf{v} = \mathbf{f}(\mathbf{y}),$$

a następnie wykorzystując prawostronną pseudoodwrotność  $\mathbf{C}^\#(\mathbf{y})$  znajdujemy

$$\mathbf{v} = \mathbf{C}^\#(\mathbf{y}) \mathbf{f}(\mathbf{y}).$$

**Krok 3.** Postulujemy sterowania układu (9.22) postaci

$$u_i^\varepsilon(\mathbf{q}, t) = \alpha_i(\mathbf{q}) + \sqrt{\frac{2}{\varepsilon}} \sum_{j=1}^l \left( \beta_i^j(\mathbf{q}) \sin\left(\frac{j t}{\varepsilon}\right) + \gamma_i^j(\mathbf{q}) \cos\left(\frac{j t}{\varepsilon}\right) \right), \quad (9.23)$$

$i = 1, \dots, m$ , których współczynniki funkcyjne  $\alpha$ ,  $\beta$ ,  $\gamma$  będą wyliczone dalej. Gdy parametr  $\varepsilon \rightarrow 0$ , sterowania stają się wysokoczęstotliwościowe i układ (9.22) jest aproksymowany z dowolnie małym błędem (zależnym od parametru  $\varepsilon$ ) równaniem

$$\dot{\mathbf{y}} = \sum_{i=1}^m \mathbf{g}_i \delta_i(\mathbf{y}) + \sum_{\substack{i,j=1 \\ i < j}}^m [\mathbf{g}_i, \mathbf{g}_j] \eta_{ij}(\mathbf{y}), \quad (9.24)$$

gdzie

$$\delta_i = \alpha_i + \sum_{j=1}^m \sum_{k=1}^l \frac{1}{k} \left( \langle d\beta_i^k, \mathbf{g}_j \rangle \gamma_j^k - \langle d\gamma_i^k, \mathbf{g}_j \rangle \beta_j^k \right) \operatorname{sgn}(j-i), \quad (9.25)$$

oraz  $d\xi = \left( \frac{\partial \xi}{\partial q_1}, \dots, \frac{\partial \xi}{\partial q_n} \right)^T$ . Współczynniki  $\boldsymbol{\eta} = (\eta_{12}, \dots, \eta_{1m}, \eta_{23}, \dots, \eta_{2m}, \dots, \eta_{m-1,m})^T \in \mathbb{R}^{m(m-1)/2}$  uzyskujemy dokonując podstawienia szybkozmiennnej składowej sterowań (9.23) do zależności (9.21), w postaci skośnie symetrycznej macierzy

$$[\boldsymbol{\eta}] = \begin{bmatrix} 0 & \eta_{12} & \cdots & \eta_{1m} \\ -\eta_{12} & 0 & \cdots & \eta_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ -\eta_{1m} & -\eta_{2m} & \cdots & 0 \end{bmatrix} = \sum_{k=1}^l \frac{1}{k} (\boldsymbol{\beta}^k \boldsymbol{\gamma}^{kT} - \boldsymbol{\gamma}^k \boldsymbol{\beta}^{kT}), \quad (9.26)$$

gdzie  $\boldsymbol{\beta}^k = (\beta_1^k, \dots, \beta_m^k)^T$ ,  $\boldsymbol{\gamma}^k = (\gamma_1^k, \dots, \gamma_m^k)^T$ .

Równanie (9.24) można odczytać jako rozwinięcie pochodnej holonomicznej trajektorii zadanej  $\mathbf{y}(t)$  względem czasu w odpowiednik szeregu Taylora, w którym w roli jednomianowych funkcji bazowych występują kolejne warstwy nawiasów Liego. Rozwijanie pochodnej trajektorii jest spowodowane prędkościowymi interpretacjami pól wektorowych. Zauważmy, że obcięcie szeregu na warstwie drugiej jest konsekwencją założenia o pełności rzędu macierzy  $\mathbf{C}(\mathbf{y})$ , złożonej z pól wektorowych pierwszego i drugiego stopnia.

**Krok 4.** Z układu równań

$$\begin{cases} \delta_i = v_i \\ [\boldsymbol{\eta}] = \begin{bmatrix} 0 & v_{m+1} & \cdots & v_{2m-1} \\ -v_{m+1} & 0 & \cdots & v_{3m-2} \\ \vdots & \vdots & \ddots & \vdots \\ -v_{2m+1} & -v_{3m-2} & \cdots & 0 \end{bmatrix} \end{cases} \quad (9.27)$$

są wyliczane funkcje  $\boldsymbol{\alpha}$ ,  $\boldsymbol{\beta}$ ,  $\boldsymbol{\gamma}$  potrzebne do określenia sterowań.

Metodę uśredniania można stosować także do układów, które dla spełnienia warunku pełnego rzędu macierzy  $\mathbf{C}(\mathbf{y})$  wymagają wzięcia więcej niż dwóch warstw algebry Liego układu. Ceną jest komplikacja procedury wyznaczania sterowań. Algorytm bazujący na technice uśredniania zilustrujemy przykładem.

### Przykład 9.3.1 (Jednokołowy robot mobilny)

Wyliczymy sterowanie z wykorzystaniem metody uśredniania dla jednokołowego robota mobilnego opisanego równaniami (8.8), o dwóch sterowaniach ( $m = 2$ ) i trójwymiarowej przestrzeni stanu ( $n = 3$ ). Algebrę Liego tego robota wyliczyliśmy w przykładzie 8.2.1. Jak łatwo sprawdzić, macierz

$$\mathbf{C}(\mathbf{y}) = [\mathbf{g}_1, \mathbf{g}_2, [\mathbf{g}_1, \mathbf{g}_2]](\mathbf{y}) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ \sin \theta & 0 & -\cos \theta \\ 0 & 1 & 0 \end{bmatrix}$$

jest pełnego rzędu ( $\det(\mathbf{C}(\mathbf{y})) = 1$ ). Niech punktem docelowym będzie konfiguracja  $\mathbf{q}_d = (0, 0, 0)^T$ . Trajektoria wiodąca do niej od punktu początkowego  $\mathbf{q}_0 = (x_0, y_0, \theta_0)^T$  zadana jest zależnością

$$\begin{pmatrix} x(t) \\ y(t) \\ \theta(t) \end{pmatrix} = \begin{pmatrix} \exp(-t)x_0 \\ \exp(-t)y_0 \\ \exp(-t)\theta_0 \end{pmatrix},$$

i spełnia równanie różniczkowe

$$\begin{cases} \dot{x} = -x \\ \dot{y} = -y \\ \dot{\theta} = -\theta. \end{cases}$$

Wyliczmy sterowania rozszerzone  $\mathbf{v}$

$$\mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \mathbf{C}^T \begin{pmatrix} -x \\ -y \\ -\theta \end{pmatrix} = \begin{pmatrix} -(x \cos \theta + y \sin \theta) \\ -\theta \\ -x \sin \theta + y \cos \theta \end{pmatrix},$$

gdzie  $\mathbf{C}$  jest takie, że  $\mathbf{C}\mathbf{C}^T = \mathbb{I}_3$ . Sterowania oryginalne wybieramy w postaci ( $l = 1$ )

$$\begin{cases} u_1^\varepsilon(t) = \alpha_1 + \sqrt{\frac{2}{\varepsilon}} \left( \beta_1^1 \sin\left(\frac{t}{\varepsilon}\right) + \gamma_1^1 \cos\left(\frac{t}{\varepsilon}\right) \right) \\ u_2^\varepsilon(t) = \alpha_2 + \sqrt{\frac{2}{\varepsilon}} \left( \beta_2^1 \sin\left(\frac{t}{\varepsilon}\right) + \gamma_2^1 \cos\left(\frac{t}{\varepsilon}\right) \right). \end{cases}$$

Łatwo sprawdzić, że równania (9.25), (9.26), (9.27) są dla rozpatrywanego robota następujące:

$$\begin{cases} \alpha_1 + \langle d\beta_1^1, \mathbf{g}_1 \rangle \gamma_1^1 - \langle d\beta_1^1, \mathbf{g}_2 \rangle \gamma_2^1 - \langle d\gamma_1^1, \mathbf{g}_1 \rangle \beta_1^1 + \langle d\gamma_1^1, \mathbf{g}_2 \rangle \beta_2^1 = v_1 \\ \alpha_2 - \langle d\beta_2^1, \mathbf{g}_1 \rangle \gamma_1^1 + \langle d\beta_2^1, \mathbf{g}_2 \rangle \gamma_2^1 + \langle d\gamma_2^1, \mathbf{g}_1 \rangle \beta_1^1 - \langle d\gamma_2^1, \mathbf{g}_2 \rangle \beta_2^1 = v_2 \\ \begin{pmatrix} \beta_1^1 \\ \beta_2^1 \end{pmatrix} (\gamma_1^1, \gamma_2^1) - \begin{pmatrix} \gamma_1^1 \\ \gamma_2^1 \end{pmatrix} (\beta_1^1, \beta_2^1) = \\ = \begin{bmatrix} 0 & \beta_1^1 \gamma_2^1 - \beta_2^1 \gamma_1^1 \\ -(\beta_1^1 \gamma_2^1 - \beta_2^1 \gamma_1^1) & 0 \end{bmatrix} = \begin{bmatrix} 0 & v_3 \\ -v_3 & 0 \end{bmatrix}. \end{cases} \quad (9.28)$$

Założmy  $\gamma_1^1 = \beta_2^1 = 0$ , by uprościć równania (9.28)

$$\begin{cases} \alpha_1 - \langle d\beta_1^1, \mathbf{g}_2 \rangle \gamma_2^1 = v_1 = -(x \cos \theta + y \sin \theta) \\ \alpha_2 + \langle d\gamma_2^1, \mathbf{g}_1 \rangle \beta_1^1 = v_2 = -\theta \\ \beta_1^1 \gamma_2^1 = v_3 = -x \sin \theta + y \cos \theta. \end{cases} \quad (9.29)$$

Wybierając  $\alpha_2 = -\theta$  i  $\gamma_2^1 = 1$  spełnimy drugie równanie układu (9.29). Z równania trzeciego wyznaczamy  $\beta_1^1 = -x \sin \theta + y \cos \theta$ . Po wykonaniu obliczeń pomocniczych

$$\begin{aligned} d\beta_1^1 &= \left( \frac{\partial \beta_1^1}{\partial x}, \frac{\partial \beta_1^1}{\partial y}, \frac{\partial \beta_1^1}{\partial \theta} \right)^T = (-\sin \theta, \cos \theta, -(x \cos \theta + y \sin \theta))^T, \\ \langle d\beta_1^1, \mathbf{g}_2 \rangle &= \langle (-\sin \theta, \cos \theta, -(x \cos \theta + y \sin \theta))^T, (0, 0, 1)^T \rangle, \end{aligned}$$

otrzymujemy współczynnik  $\alpha_1 = -2(x \cos \theta + y \sin \theta)$ . Podsumowując wyniki częściowe, uzyskujemy sterowania w postaci

$$\begin{cases} u_1^\varepsilon = -2(x \cos \theta + y \sin \theta) + \sqrt{\frac{2}{\varepsilon}} \left( (-x \sin \theta + y \cos \theta) \sin\left(\frac{t}{\varepsilon}\right) \right) \\ u_2^\varepsilon = -\theta + \sqrt{\frac{2}{\varepsilon}} \cos\left(\frac{t}{\varepsilon}\right). \end{cases}$$

Krytycy metody uśredniania zwracają uwagę na jej dużą złożoność obliczeniową\* oraz konieczność zastosowania sterowań wysokoczęstotliwościowych†. ■

## 9.4 Metoda Lie-algebraiczna

Do lokalnych metod planowania ruchu robotów nieholonomicznych należy *metoda Lie-algebraiczna* bazująca na algebrze Liego układu. Polega ona na iteracyjnym generowaniu pól wektorowych (kierunków ruchu) w bieżącym punkcie przestrzeni stanu, a następnie realizacji tych kierunków przy pomocy sterowań. Kierunki są tworzone przez obliczanie wartości początkowych elementów bazy Halla w bieżącym punkcie przestrzeni stanu. Lokalność metody polega na tym, że ruch w kierunku nawiasów Liego zachodzi tylko dla infinitezymalnych przemieszczeń. Zaletą metody algebraicznej jest jej konstruktywność i ogólność. W specjalnych wersjach metoda umożliwia także optymalizację lokalnych kryteriów jakości. Podstawową własność zbieżności stanu bieżącego do stanu docelowego zapewnia poniższy lemat, będący prostą konsekwencją twierdzenia Chow.

**Lemat 9.4.1** *Jeżeli układ nieholonomiczny spełnia warunek rzędu algebry Liego układu, to istnieje taki kierunek  $\mathbf{X}(\mathbf{q}_0)$  powstały przez obliczenie wartości jednego z elementów bazy Halla (możliwie niskiego stopnia) w bieżącym punkcie przestrzeni stanu  $\mathbf{q}_0$  i taka chwila czasu  $t$ , że przemieszczenie w kierunku  $\mathbf{X}(\mathbf{q}_0)$  do chwili  $t$  zapewnia zmniejszenie odległości euklidesowej do stanu docelowego  $\mathbf{q}_d$ .*

Lemat ten zaleca wybór kierunku zmniejszającego odległość do celu spośród elementów bazy Halla niskiego stopnia, gdyż pola niższych stopni są łatwiejsze do generacji przy pomocy sterowań i mniej kosztowne energetycznie\*.

\*Widoczną zwłaszcza gdy  $m = 2$ , a  $n \geq 6$ .

†Gdy  $\varepsilon \rightarrow 0$ .

\*Zobacz wniosek z przykładu 8.2.5.

Interpretacja geometryczna lematu jest następująca. Jeżeli kierunek pola wektorowego będącego elementem bazy Halla nie jest prostopadły w punkcie  $\mathbf{q}_0$  do kierunku  $\mathbf{q}_d - \mathbf{q}_0$ , to ruch w tym kierunku (choćby przez krótki czas) zapewnia zmniejszenie odległości do stanu docelowego. Jeżeli kierunek ten tworzy z wektorem  $\mathbf{q}_d - \mathbf{q}_0$  kąt większy od  $\pi/2$ , to przez odpowiednią modyfikację sterowań możliwe jest wygenerowanie kierunku przeciwnego, tworzącego z kierunkiem ku celowi kąt mniejszy od  $\pi/2$ . Taki „kierunek poprawy” zawsze istnieje, gdyż nie jest możliwe, by wszystkie kierunki uzyskane przez ewaluację bazy Halla w  $\mathbf{q}_0$  były prostopadłe do  $\mathbf{q}_d - \mathbf{q}_0$ <sup>†</sup>.

Sposób generacji kierunku przeciwnego do zadanego jest następujący. Załóżmy, że sterowanie  $\mathbf{u}(t)$ ,  $t \in [0, T]$  w pewnym punkcie przestrzeni stanu  $\mathbf{q}_0$  generuje infinitezymalne przemieszczenie wzdłuż pewnego pola  $\mathbf{Z}$ , tj.  $\exp(t\mathbf{Z})\mathbf{q}_0 \simeq \mathbf{q}_0 + t\mathbf{Z}(\mathbf{q}_0)$ . Pokażemy, że sterowanie  $\tilde{\mathbf{u}}(t) = -\mathbf{u}(T - t)$ ,  $t \in [0, T]$  generuje kierunek  $\mathbf{W}(\mathbf{q}_0)$  przeciwny do  $\mathbf{Z}(\mathbf{q}_0)$ . Z własności złożenia strumieni generowanych sterowaniami  $\mathbf{u}$ ,  $\tilde{\mathbf{u}}$  otrzymujemy

$$\exp(t\mathbf{W}) \exp(t\mathbf{Z})\mathbf{q}_0 = \mathbf{q}_0. \quad (9.30)$$

Stosując formułę CBHD do lewej strony (9.30) i poprzestając na składnikach liniowych uzyskujemy

$$\exp(t\mathbf{W}) \exp(t\mathbf{Z})\mathbf{q}_0 \simeq \exp(t\mathbf{W} + t\mathbf{Z})\mathbf{q}_0 \simeq t\mathbf{W}(\mathbf{q}_0) + t\mathbf{Z}(\mathbf{q}_0) + \mathbf{q}_0 \simeq \mathbf{q}_0.$$

co dowodzi, że dla krótkich czasów  $t$  kierunek  $\mathbf{W}(\mathbf{x}_0) = -\mathbf{Z}(\mathbf{x}_0)$ .

Konstrukcję bazy Halla przedstawiono w rozdziale 8. Dla układu sterowanego w zadanym punkcie przestrzeni stanu można w łatwy sposób wyznaczyć wektor odpowiadający elementowi bazy Halla. Istotnym problemem jest natomiast realizacja zadanego kierunku przy pomocy sterowań. Do tego celu wykorzystamy uogólnioną formułę *Campbella-Bakera-Hausdorffa-Dynkina*, a następnie przedstawimy sekwencję kroków algorytmu realizującego metodę Lie-algebraiczną planowania ruchu układów nieholonomicznych. W podrozdziale 9.4.2 zilustrujemy metodę Lie-algebraiczną na przykładzie planowania ruchu jednokołowego robota mobilnego. W podrozdziale 9.4.3 pokażemy, że uogólniona formuła CBHD unifikuje różne metody planowania ruchu robotów nieholonomicznych.

<sup>†</sup>Nie byłyby wówczas spełniony warunek rzędu.

### 9.4.1 Uogólniona formuła Campbella-Bakera-Hausdorffa-Dynkina

*Uogólniona formuła Campbella-Bakera-Hausdorffa-Dynkina* (UCBHD) opisuje (lokalnie) trajektorię nieautonomicznego układu równań różniczkowych

$$\dot{\mathbf{q}} = \mathbf{F}(t)(\mathbf{q}) \quad (9.31)$$

z zadanym warunkiem początkowym  $\mathbf{q}(0) = \mathbf{q}_0$ , gdzie  $\mathbf{F}(t)(\cdot)$  jest rodziną analitycznych pól wektorowych zależnych od czasu. Rozwiązanie (9.31) jest postaci

$$\mathbf{q}(t) = \exp(\mathbf{z}(t))(\mathbf{q}_0)|_{s=1}. \quad (9.32)$$

Dla czasów  $t$  bliskich zera,  $\mathbf{z}(t)$  wyraża się szeregiem

$$\mathbf{z}(t)(\mathbf{q}) = \sum_{r=1}^{\infty} \sum_{\sigma \in \mathbb{P}_r} \frac{(-1)^{err(\sigma)}}{r^2 \binom{r-1}{err(\sigma)}} \int_{\mathbb{T}_r(t)} [\cdots [\mathbf{F}(s_{\sigma(1)}), \mathbf{F}(s_{\sigma(2)})] \cdots], \mathbf{F}(s_{\sigma(r)})] \mathbf{ds}, \quad (9.33)$$

gdzie wartości  $\mathbf{F}(s)$  wyciżamy w punkcie  $\mathbf{q}$ . W nieautonomicznym układzie równań różniczkowych (9.31) jest wymagana analityczność pól wektorowych, bowiem nieskończony szereg pól wektorowych (9.33) musi być zbieżny. Pozostałe symbole występujące w (9.33) mają następujące znaczenie:

$\mathbb{P}_r$  jest zbiorem permutacji elementów  $\{1, \dots, r\}$ , np.  $\mathbb{P}_3 = \{(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1)\}$ .

$err(\sigma)$ ,  $\sigma \in \mathbb{P}_r$  jest liczbą nieporządków w permutacji:  $\sigma = \{\sigma(1), \sigma(2), \dots, \sigma(r-1), \sigma(r)\}$ . Liczba nieporządków rośnie o jeden, gdy kolejny element permutacji jest mniejszy od bieżącego elementu\*.

$\mathbb{T}_r(t)$  jest  $r$ -wymiarowym sympleksem,  $\mathbb{T}_r(t) = \{s \in \mathbb{R}^r | 0 < s_1 < s_2 < \dots < s_r < t\}$ .

Uogólnioną formułę CBHD zinterpretujemy jako operator zmiany stanu, który dla krótkich czasów  $t$  przemieszcza stan bieżący  $\mathbf{q}_0$  układu (9.31) do stanu  $\mathbf{q}_0 + \mathbf{z}(t)(\mathbf{q}_0)$ . Zatem,  $\mathbf{z}(t)(\mathbf{q}_0)$  jest wektorem translacji stanu, czyli kierunkiem ruchu w punkcie  $\mathbf{q}_0$ . Formułę (9.33) można wykorzystać bądź do opisu translacji stanu, gdy zadane są sterowania, bądź do wyznaczenia sterowań, gdy jest zadane infinytezymalne przemieszczenie stanu (kierunek

\*Np.  $err((1, 2, 3)) = 0$ ,  $err((1, 3, 2)) = 1$ ,  $err((3, 2, 1)) = 2$ .

ruchu). Ten drugi aspekt uogólnionej formuły CBHD wykorzystamy w planowaniu ruchu robotów nieholonomicznych opisanych równaniami

$$\dot{\mathbf{q}} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{q})\mathbf{u}_i(t) = \mathbf{F}(t)(\mathbf{q}), \quad (9.34)$$

z analitycznymi polami wektorowymi  $\mathbf{g}_i$ , stanem  $\mathbf{q} \in \mathbb{R}^n$  i sterowaniami  $\mathbf{u}(t) \in \mathcal{U} \subset \mathbb{R}^m$ ,  $m < n$ . Zakładamy, że kierunek ruchu  $\mathbf{X}(\mathbf{p})$  w bieżącym punkcie przestrzeni stanu  $\mathbf{p}$  jest znany, natomiast należy wyznaczyć sterowania realizujące ten kierunek przez krótki czas  $t$ . Pożądane przemieszczenie stanu  $\mathbf{X}(\mathbf{p})$  przedstawiamy jako kombinację liniową początkowych elementów bazy Halla w punkcie  $\mathbf{p}$

$$t\mathbf{X}(\mathbf{p}) = \sum_i \alpha_i(t)\mathbf{H}_i(\mathbf{p}),$$

gdzie elementy bazy Halla  $\mathbf{H}_i$  (pola wektorowe) są niezależne w tymże punkcie. Podobnie jak w lemacie 9.4.1, preferowanie początkowych elementów bazy Halla jest spowodowane energetycznie efektywniejszym i wymagającym mniejszej liczby przełączeń sposobem generacji pól wektorowych niskich stopni przy pomocy sterowań. W ten sposób uzyskujemy przedstawienie zadanego przemieszczenia w bazie Halla. Teraz należy wyznaczyć sterowania realizujące tę kombinację elementów bazy Halla. Podstawiając prawą stronę modelu kinematyki (9.34) do ogólnej zależności na infinitesimalne przemieszczenie stanu (9.33) i wykorzystując skośną symetrię nawiasów Liego oraz tożsamość Jacobiego, uzyskujemy wyrażenie na przemieszczenie stanu w funkcji sterowań

$$\mathbf{z}(t)(\mathbf{p}) = \sum_i \beta_i(t, \mathbf{u}(\cdot))\mathbf{H}_i(\mathbf{p}),$$

gdzie  $\beta_i$  są pewnymi funkcjonalami zależnymi wyłącznie od sterowań. Dla zadanego przemieszczenia  $\mathbf{z}(t)(\mathbf{p}) = t\mathbf{X}(\mathbf{p})$  uzyskujemy układ równań całkowych

$$\alpha_i(t) = \beta_i(t, \mathbf{u}(\cdot)), \quad i = 1, \dots \quad (9.35)$$

Aby rozwiązać układ równań (9.35) przekształcamy go do układu równań algebraicznych. W tym celu wybieramy sterowania  $u_i(\cdot)$ ,  $i = 1, \dots, m$ , w postaci pewnej sparametryzowanej rodziny funkcji i uzyskujemy równania algebraiczne, w których zmiennymi do wyznaczenia są parametry funkcji.



**Przykład 9.4.1 (Układ dwuwęściowy)**

Wyliczmy sterowania przeprowadzające stan układu w kierunku najprostszego złożonego pola wektorowego układu dwuwęściowego

$$\dot{\mathbf{q}}(t) = \mathbf{X}(\mathbf{q})\mathbf{u}(t) + \mathbf{Y}(\mathbf{q})\mathbf{v}(t) = \mathbf{F}(t)(\mathbf{q}) \quad (9.36)$$

ze sterowaniami  $\mathbf{u}(t)$ ,  $\mathbf{v}(t)$  i analitycznymi polami wektorowymi  $\mathbf{X}$ ,  $\mathbf{Y}$ .

Naszym zadaniem jest wyznaczenie sterowań  $\mathbf{u}(s)$  i  $\mathbf{v}(s)$ ,  $s \in [0, t]$  realizujących (infinitesimalny) ruch w kierunkach pól wektorowych  $\mathbf{X}$ ,  $\mathbf{Y}$ ,  $[\mathbf{X}, \mathbf{Y}]$ . Generacja kierunków  $\mathbf{X}$ ,  $\mathbf{Y}$  jest natychmiastowa:  $\mathbf{u}(s) \equiv 1$ ,  $\mathbf{v}(s) \equiv 0$  dla  $\mathbf{X}$ , oraz  $\mathbf{u}(s) \equiv 0$ ,  $\mathbf{v}(s) \equiv 1$  dla  $\mathbf{Y}$ . Generacja  $[\mathbf{X}, \mathbf{Y}]$  jest znacznie trudniejsza. Stosując uogólnioną formułę CBHD otrzymujemy

$$\begin{aligned} \mathbf{z}(t) = \int_0^t \mathbf{F}(s_1) ds^1 + \frac{1}{2^2} \int_{\mathbb{T}_2(t)} [\mathbf{F}(s_1), \mathbf{F}(s_2)] ds^2 + \\ - \frac{1}{2^2} \int_{\mathbb{T}_2(t)} [\mathbf{F}(s_2), \mathbf{F}(s_1)] ds^2 + \dots \quad (9.37) \end{aligned}$$

W powyższym wzorze zostały zastosowane następujące oznaczenia:  $ds^r = ds_1 ds_2 ds_3 \dots ds_r$ ,  $\int_{\mathbb{T}_r(t)} = \int_0^t \int_0^{s^r} \dots \int_0^{s^2}$ .

Zanim podstawimy do wzoru (9.37) pole  $\mathbf{F}(t)$  zadane równaniem (9.36), zamieńmy przedział  $[0, t] \ni s$  określoności sterowań  $\mathbf{u}(s)$ ,  $\mathbf{v}(s)$  na unormowany przedział  $[0, 1]$ . W tym celu przeskalujemy zmienną opisującą czas  $s$ , na zmienną czasową  $\check{s}$  kładąc

$$\check{s} = \frac{s}{t}.$$

Po przeskalowaniu, całkowanie odbywa się po sympleksie  $\mathbb{T}_r(1)$  zamiast po  $\mathbb{T}_r(t)$ , a sterowania  $\mathbf{u}(s)$ ,  $\mathbf{v}(s)$  stają się zależne od parametru  $t$ , tj. przyjmują postać  $\mathbf{u}(\check{s}, t)$ ,  $\mathbf{v}(\check{s}, t)$ . Oczywiście,  $ds^r = t^r d\check{s}^r$ . Zdefiniujemy sterowania skalowane jako

$$\begin{cases} \check{\mathbf{u}}(\check{s}, t) = t\mathbf{u}(\check{s}, t) \\ \check{\mathbf{v}}(\check{s}, t) = t\mathbf{v}(\check{s}, t). \end{cases}$$

W dalszym ciągu znak  $\sim$  nad  $\mathbf{u}$ ,  $\mathbf{v}$ ,  $s$  będziemy opuszczać. Przeskalowanie sterowań jest operacją odwracalną i wskazuje, że można regulować czas sterowania przez modyfikację amplitud sterowania i na odwrót — można wpływać na wielkość sterowań przez odpowiedni dobór czasu sterowania. Własność ta jest szczególnie ważna, gdy sterowania są ograniczone.

Podstawiając równania systemu (9.36) do (9.37) uzyskuje się wyrażenie

$$\mathbf{z}(t) = \mathbf{X} \int_0^1 u_1 ds_1 + \mathbf{Y} \int_0^1 v_1 ds_1 + \frac{1}{2} \int_{\mathbb{T}_2(1)} [\mathbf{X}u_1 + \mathbf{Y}v_1, \mathbf{X}u_2 + \mathbf{Y}v_2] ds^2 + \dots, \quad (9.38)$$

gdzie  $u_i = u(s_i, t)$ ,  $v_j = v(s_j, t)$ ,  $s_i, s_j \in [0, 1]$ . Wykorzystując dwuliniowość nawiasu Liego upraszczamy równanie (9.38)

$$\mathbf{z}(t) = \mathbf{X} \int_0^1 u_1 ds_1 + \mathbf{Y} \int_0^1 v_1 ds_1 + \frac{1}{2} [\mathbf{X}, \mathbf{Y}] \int_{\mathbb{T}_2(1)} (u_1 v_2 - u_2 v_1) ds^2 + \dots \quad (9.39)$$

Zauważmy, że (9.39) ma postać

$$\mathbf{z}(t) = \mathbf{X}\beta_{\mathbf{X}}(t) + \mathbf{Y}\beta_{\mathbf{Y}}(t) + [\mathbf{X}, \mathbf{Y}]\beta_{[\mathbf{X}, \mathbf{Y}]}(t) + \dots \quad (9.40)$$

ze współczynnikami zależnymi od sterowań, danymi równaniami

$$\begin{aligned} \beta_{\mathbf{X}}(t) &= \int_0^1 u(s, t) ds, & \beta_{\mathbf{Y}}(t) &= \int_0^1 v(s, t) ds, \\ \beta_{[\mathbf{X}, \mathbf{Y}]}(t) &= \frac{1}{2} \int_{\mathbb{T}_2(1)} (u_1 v_2 - u_2 v_1) ds^2, \\ \beta_{[\mathbf{X}, [\mathbf{X}, \mathbf{Y}]]}(t) &= \dots, & \beta_{[\mathbf{Y}, [\mathbf{X}, \mathbf{Y}]]}(t) &= \dots, \dots \end{aligned} \quad (9.41)$$

Zbiór równań całkowych (9.41) rozwiążemy przyjmując sterowania sinusoidalne. Zadany kierunek  $t[\mathbf{X}, \mathbf{Y}]$  wygenerujemy sterowaniami

$$\begin{cases} u(s, t) = \bar{u}(t) \sin(a_1 s + b_1) \\ v(s, t) = \bar{v}(t) \sin(a_2 s + b_2), \end{cases}$$

gdzie  $\bar{u}(t)$ ,  $\bar{v}(t)$  oznaczają amplitudy sterowań, a  $s \in [0, 1]$ . Przyjmując częstotliwości sterowań

$$\begin{cases} a_1 = 2\pi k_1 \\ a_2 = 2\pi k_2, \end{cases}$$

z  $k_1, k_2 \in \mathbb{N}$ , uzyskujemy znikanie w (9.41) współczynników  $\beta_{\mathbf{X}}$ ,  $\beta_{\mathbf{Y}}$  odpowiadających polom  $\mathbf{X}$ ,  $\mathbf{Y}$ .

Dzięki przyjęciu sterowań sinusoidalnych, wyliczenie współczynników  $\beta$  z układu równań (9.41) odbywa się w dwóch etapach. W pierwszym z nich określone są parametry sinusoid (częstotliwości i przesunięcia fazowe)  $a_1$ ,  $b_1$ ,  $a_2$ ,  $b_2$ . Drugi etap to określenie amplitud sterowań  $\bar{u}(t)$ ,  $\bar{v}(t)$ . Rozpocznijmy od wyliczenia zależności  $\beta_{[X,Y]}$  od sterowań. Całka pierwszego składnika we współczynniku  $\beta_{[X,Y]}$  wynosi

$$\begin{aligned} \int_0^1 \int_0^{s_2} u_1 v_2 ds_1 ds_2 &= \\ &= \int_0^1 \int_0^{s_2} \sin(a_1 s_1 + b_1) \sin(a_2 s_2 + b_2) ds_1 ds_2 = \\ &= \frac{1}{2a_1} \int_0^1 \sin((a_1 - a_2)s_2 + b_1 - b_2) ds_2 + \\ &\quad - \frac{1}{2a_1} \int_0^1 \sin((a_1 + a_2)s_2 + b_1 + b_2) ds_2. \end{aligned}$$

Jeśli teraz założymy, że

$$a_1 = a_2 \quad (\text{równoważnie } k_1 = k_2), \quad (9.42)$$

to

$$\int_0^1 \int_0^{s_2} u_1 v_2 ds_1 ds_2 = \frac{1}{2a_1} \sin(b_1 - b_2).$$

Aby wyliczyć całkę iloczynu  $u_2 v_1$ , wystarczy w powyższych wyrażeniach zamienić miejscami indeksy, uzyskując tym samym

$$\int_0^1 \int_0^{s_2} u_2 v_1 ds_1 ds_2 = \frac{1}{2a_2} \sin(b_2 - b_1) = -\frac{1}{2a_2} \sin(b_1 - b_2).$$

Sumując wyliczone całki otrzymujemy ostatecznie

$$\beta_{[X,Y]}(t) = \frac{1}{2} \bar{u}(t) \bar{v}(t) \frac{a_1 + a_2}{2a_1 a_2} \sin(b_1 - b_2).$$

Uwzględniając warunek (9.42) uzyskujemy

$$\beta_{[X,Y]}(t) = \frac{1}{2a_1} \bar{u}(t) \bar{v}(t) \sin(b_1 - b_2) = \frac{1}{4\pi k_1} \bar{u}(t) \bar{v}(t) \sin(b_1 - b_2). \quad (9.43)$$

Zauważmy, że dobór faz początkowych sterowań  $b_1$ ,  $b_2$  umożliwia dowolną regulację przesunięcia fazowego  $b_1 - b_2$ . Rozsądnie jest jednak dobrać  $b_1$  i  $b_2$  tak, by ich różnica maksymalizowała wartość wyrażenia (9.43)

$$|\sin(b_1 - b_2)| = 1. \quad (9.44)$$

Warunek (9.44) nazywamy *warunkiem fazowym*. Pozwala on na zmniejszenie amplitudy sterowań  $\bar{u}$  i  $\bar{v}$ . Jedną z par faz początkowych spełniających warunek fazowy podczas generacji  $[\mathbf{X}, \mathbf{Y}]$  jest

$$b_1 = \frac{\pi}{2}, \quad b_2 = 0. \quad (9.45)$$

Spośród wielu możliwych wartości  $k_1$  i  $k_2$  spełniających (9.42) należy wybrać możliwie najmniejsze. Taki wybór zwiększa współczynnik  $\beta_{[\mathbf{X}, \mathbf{Y}]}$ , a zatem amplitudy sterowań zapewniające stałą wartość  $\beta_{[\mathbf{X}, \mathbf{Y}]}$  są mniejsze. Powiadamy, że przy generacji  $t[\mathbf{X}, \mathbf{Y}]$  jest spełniony *warunek częstotliwościowy*, gdy

$$k_1 = k_2 = 1. \quad (9.46)$$

Ponieważ ruch w kierunku  $[\mathbf{X}, \mathbf{Y}]$  ma trwać przez czas  $t$ , zatem wykorzystując równania (9.43), (9.45) i (9.46) uzyskujemy *warunek amplitudowy*

$$\beta_{[\mathbf{X}, \mathbf{Y}]}(t) = t = \frac{1}{4\pi} \bar{u}(t) \bar{v}(t). \quad (9.47)$$

Minimalizacja energii sterowań

$$\begin{aligned} \int_0^1 (u^2(s, t) + v^2(s, t)) ds &= \\ &= \int_0^1 (\bar{u}^2(t) \sin^2(a_1 s + b_1) + \bar{v}^2(t) \sin^2(a_2 s + b_2)) ds = \\ &= \frac{\bar{u}^2(t) + \bar{v}^2(t)}{2} \end{aligned}$$

jest zatem równoważna minimalizacji  $\bar{u}^2(t) + \bar{v}^2(t)$ . Warunek (9.47) jest spełniony dla jednakowych amplitud sterowań  $\bar{u}(t) = \bar{v}(t)$ . Ostatecznie, uwzględniając warunek częstotliwościowy (9.46), fazowy (9.45) i amplitudowy (9.47), uzyskujemy sterowania generujące pole  $t[\mathbf{X}, \mathbf{Y}]$  w postaci

$$\begin{aligned} u(t, s) &= 2\sqrt{\pi}\sqrt{t} \cos(2\pi s), \\ v(t, s) &= 2\sqrt{\pi}\sqrt{t} \sin(2\pi s), \quad s \in [0, 1]. \end{aligned} \quad (9.48)$$

Zauważmy dodatkowo, że po podstawieniu do (9.41) sterowań zadanych zależnością (9.48) nie uzyskujemy automatycznie zerowania współczynników  $\beta_{[\mathbf{X}, [\mathbf{X}, \mathbf{Y}]]}(t)$ ,  $\beta_{[\mathbf{Y}, [\mathbf{X}, \mathbf{Y}]]}(t)$ , itp., ale nawiasy Liego wyższych stopni zaburzające pole mają amplitudę zależną od  $t$  w taki sposób, że dla krótkich  $t$  są pomijalne w porównaniu z efektem pola  $[\mathbf{X}, \mathbf{Y}]$ . ■

### 9.4.2 Algorytm metody Lie-algebraicznej

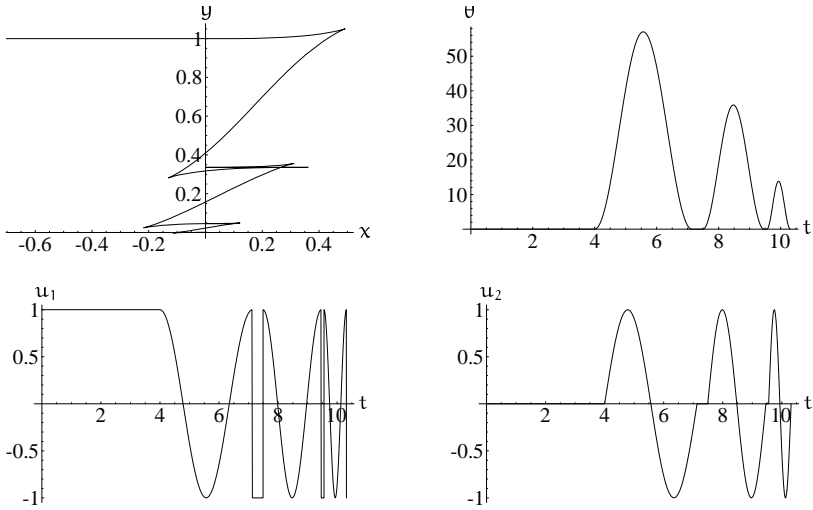
Metoda Lie-algebraiczna planuje ruch układu nieholonomicznego opisanego równaniami  $\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u}$  z punktu początkowego  $\mathbf{q}_0$  do punktu końcowego  $\mathbf{q}_d$  w środowisku bezkolizyjnym. Metoda może być zaimplementowana stosownie do następującego algorytmu.

- Krok 1.** Wygenerować pierwszych  $N \geq n$  elementów bazy Halla  $\mathbf{H} = \{\mathbf{H}_1, \mathbf{H}_2, \dots, \mathbf{H}_N\}$ , które w każdym punkcie przestrzeni stanu rozpinają  $n$ -wymiarową przestrzeń stanu.
- Krok 2.** Wprowadzić punkt początkowy  $\mathbf{q}_0$  i punkt docelowy  $\mathbf{q}_d$  zadania planowania ruchu. Podstawić za punkt bieżący  $\mathbf{q}_c$  punkt początkowy.
- Krok 3.** Sprawdzić warunek osiągnięcia stanu docelowego  $\|\mathbf{q}_c - \mathbf{q}_d\| < \varepsilon$ , gdzie  $\varepsilon$  jest zadaną dokładnością osiągnięcia stanu docelowego. Jeśli warunek jest spełniony algorytm kończy działanie, jeśli nie — przechodzi do kolejnego kroku.
- Krok 4.** Przeglądając kolejno elementy zbioru  $\mathbf{H}$  wyznaczyć pierwszy z wektorów kierunków  $\mathbf{H}_i(\mathbf{q}_c)$ , który nie jest prostopadły do kierunku  $\mathbf{q}_d - \mathbf{q}_c$ .
- Krok 5.** Wyznaczyć sterowania realizujące kierunek  $\mathbf{H}_i(\mathbf{q}_c)$  (ewentualnie zmieniając kierunek  $\mathbf{H}_i(\mathbf{q}_c)$  na przeciwny), by zapewnić zmniejszenie odległości do stanu docelowego. Czas ruchu wzdłuż kierunku  $\pm\mathbf{H}_i(\mathbf{q}_c)$  dobrać tak, by zmniejszenie odległości było jak największe (nawet za cenę pogorszenia realizacji kierunku  $\pm\mathbf{H}_i(\mathbf{q}_c)$ ). Osiągnięty stan staje się nowym stanem bieżącym  $\mathbf{q}_c$ .
- Krok 6.** Powrócić do kroku 3.

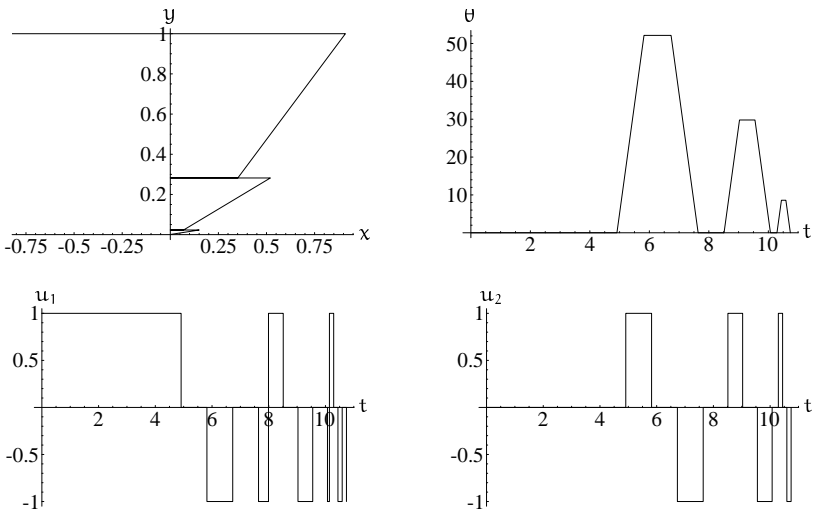
#### Przykład 9.4.2 (Jednokołowy robot mobilny)

Powyższy algorytm został przetestowany dla modelu jednokołowego robota mobilnego. Zadanie planowania ruchu polegało na przemieszczeniu robota od stanu początkowego  $\mathbf{q}_0 = (x_0, y_0, \theta_0)^T = (-4, 1, 0)^T$  do stanu docelowego  $\mathbf{q}_d = (x_d, y_d, \theta_d)^T = (0, 0, 0)^T$ . Jak pokazaliśmy w rozdziale 9.3, pola wektorowe  $\mathbf{g}_1, \mathbf{g}_2, [\mathbf{g}_1, \mathbf{g}_2]$  rozpinają przestrzeń stanu w każdym punkcie, zatem  $N = 3$ . Trajektorię na płaszczyźnie  $\times Y$ , zmienność orientacji robota w czasie, oraz przebiegi sterowań realizujących zadanie przedstawiono na rysunkach 9.5 i 9.6\*. Wykresy z rysunku 9.5 ilustrują przebiegi, gdy ge-

\*W przykładach prezentowanych w tym rozdziale obliczenia wykonano w środowisku MATHEMATICA®.



**Rysunek 9.5** Trajektoria na płaszczyźnie  $XY$ , zmiana kąta  $\theta$  w czasie oraz sterowania odcinkami ciągłe (uzyskane uogólnioną formułą CBHD) realizujące zadanie planowania ruchu dla jednokołowego robota mobilnego.



**Rysunek 9.6** Trajektoria na płaszczyźnie  $XY$ , zmiana kąta  $\theta$  w czasie oraz sterowania odcinkami stałe (uzyskane przy pomocy formuły CBHD) realizujące zadanie planowania ruchu dla jednokołowego robota mobilnego.

nerowane pole jest realizowane sterowaniami ciągłymi (formuła uogólniona CBHD), natomiast na rysunku 9.6 zostały zamieszczone przebiegi realizowane przy pomocy sterowań odcinkami stałych (formuła CBHD). Zauważmy, że sterowania ciągłe są łatwiejsze do realizacji praktycznej, ze względu na ograniczenie liczby przełączeń sterowań tylko do momentów zmiany pola wyznaczającego lokalny kierunek ruchu. ■

### 9.4.3 Znaczenie uogólnionej formuły Campbella-Bakera-Hausdorffa-Dynkina

Uogólniona formuła CBHD pozwala na unifikację wielu metod planowania ruchu robotów nieholonomicznych. Z tej formuły wynika bowiem poniższy lemat, ważny dla krótkich czasów  $t$ .

**Lemat 9.4.2** *Jeżeli sterowania  $u_i(s)$ ,  $i = 1, \dots, m$  układu nieholonomicznego  $\dot{\mathbf{q}} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{q})u_i$ ,  $\mathbf{q} \in \mathbb{R}^n$  dla małej wartości parametru  $t$  spełniają warunek*

$$\forall i = 1, \dots, m \quad \int_0^t u_i(s) ds = 0, \quad (9.49)$$

*to uzyskujemy ruch w kierunku pól wyższych stopni, natomiast ruch w kierunku pól-generatorów jest zabroniony. Algebraicznie teza lematu oznacza, że*

$$\begin{aligned} \varphi_{\mathbf{q}_0, t}(\mathbf{u}(\cdot)) &\simeq \mathbf{q}_0 + \mathbf{z}(t)(\mathbf{q}_0) = \\ &= \mathbf{q}_0 + \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m [\mathbf{g}_i, \mathbf{g}_j](\mathbf{q}_0) \int_0^t \int_0^{s_2} u_i(s_1) u_j(s_2) ds_1 ds_2. \end{aligned} \quad (9.50)$$

Na mocy uogólnionej formuły CBHD w układzie  $\dot{\mathbf{q}} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{q})u_i(t) = \mathbf{F}(t)(\mathbf{q})$  przemieszczenie stanu dla krótkich czasów  $t$  i stanu  $\mathbf{q}_0$  wynosi

$$\begin{aligned} \mathbf{z}(t)(\mathbf{q}_0) &\simeq \int_0^t \mathbf{F}(s) ds + \frac{1}{4} \int_0^t \int_0^{s_2} [\mathbf{F}(s_1), \mathbf{F}(s_2)] ds_1 ds_2 + \\ &\quad - \frac{1}{4} \int_0^t \int_0^{s_2} [\mathbf{F}(s_2), \mathbf{F}(s_1)] ds_1 ds_2 = \\ &\quad = \int_0^t \mathbf{F}(s) ds + \frac{1}{2} \int_0^t \int_0^{s_2} [\mathbf{F}(s_1), \mathbf{F}(s_2)] ds_1 ds_2, \end{aligned}$$

gdzie wartości  $\mathbf{F}$  są wyliczane w punkcie  $\mathbf{q}_0$ . Po podstawieniu równań układu nieholonomicznego mamy

$$\mathbf{z}(t)(\mathbf{q}_0) \simeq \sum_{i=1}^m \mathbf{g}_i(\mathbf{q}_0) \int_0^t \mathbf{u}_i(s) ds + \\ + \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m [\mathbf{g}_i, \mathbf{g}_j](\mathbf{q}_0) \int_0^t \int_0^{s_2} \mathbf{u}_i(s_1) \mathbf{u}_j(s_2) ds_1 ds_2.$$

Na mocy warunku (9.49) uzyskujemy tezę lematu.

Spełnienie warunku (9.49) nie powoduje ruchu w kierunkach generatorów  $\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_m$ , natomiast, w zależności od sterowań, umożliwia ruch w kierunkach  $[\mathbf{g}_1, \mathbf{g}_2], [\mathbf{g}_1, \mathbf{g}_3], \dots, [\mathbf{g}_{m-1}, \mathbf{g}_m]$ , które mogą być wymagane do zapewnienia sterowalności układu\*. Pola wektorowe-generatory można zrealizować jednym sterowaniem stałym. Warunek (9.49) nie jest restrykcyjny, bowiem spełnia go każde sterowanie okresowe bez składowej stałej. Stąd też obecność sterowań sinusoidalnych w zadaniach planowania ruchu. Ponadto sterowania sinusoidalne pozostawiają możliwość doboru amplitudy, częstotliwości i przesunięcia fazowego sterowania, którymi można kształtować kierunek przemieszczenia wyrażony kombinacją pól stopni wyższych od 1. Powyższe rozważania są słuszne lokalnie, dla krótkich czasów  $t$ , dla wszystkich układów nieholonomicznych. Dla szczególnych podklas układów nieholonomicznych, np. łańcuchowych†, własność przemieszczania stanu w „trudnych” kierunkach bez przemieszczenia w kierunkach „łatwych” globalizuje się.

Teza lematu 9.4.2 jest efektywnie wykorzystywana przez wiele metod planowania ruchu. Wśród tych metod znajduje się metoda sterowań sinusoidalnych dla układów łańcuchowych, metoda bazująca na twierdzeniu Stokesa, metoda Lie-algebraiczna. Dwie pierwsze z wymienionych metod mają zastosowanie do układów nieholonomicznych o specjalnej strukturze modelu. Dzięki owej strukturze współrzędne wektora stanu dzielą się na dwie grupy: współrzędne sterowane bezpośrednio i sterowane pośrednio (przez zmianę współrzędnych z pierwszej grupy). Znalazienie sterowań przeprowadzających współrzędne sterowane bezpośrednio do ich wartości zadanych jest trywialne. Natomiast osiągnięcie przez współrzędne sterowane pośrednio ich docelowych wartości odbywa się poprzez kreślenie pętli

\*Generatory nie wystarczają, ponieważ  $n > m$ .

†Zobacz rozdział 10.2.



przez współrzędne sterowane bezpośrednio. Możliwość przeorientowania współrzędnych sterowanych pośrednio jest konsekwencją lematu 9.4.2.

Metoda Lie-algebraiczna wykorzystuje tezę lematu 9.4.2 w nieco inny sposób. Dzięki sterowaniom cyklicznym uzyskuje się infinitesimalne przemieszczenie w ściśle określonym kierunku, co jest warunkiem koniecznym uzyskania zbieżności metody planowania ruchu. Trudność realizacji ruchu w zadanym kierunku w  $n$ -wymiarowej przestrzeni stanu wynika z posiadania przez układ jedynie  $m$  sterowań,  $m < n$ . Sterowania okresowe nie powodują ruchu w kierunkach, które łatwiej wygenerować prostszymi sterowaniami, natomiast gwarantują ruch w kierunkach wymaganych do spełnienia warunku sterowalności układu, a w konsekwencji zapewniają rozwiązanie zadania planowania ruchu.

## 9.5 Komentarze i uwagi bibliograficzne

Sformułowanie Zasady Maksimum Pontriagina podajemy w dodatku A.5, zawierają je także klasyczne podręczniki do teorii sterowania lub teorii optymalizacji, np. [AF69, FSW80, VG97]. W [FSW80] stwierdzono, że zbieżność schematu obliczeniowego zadania dwubrzegowego generowanego przez Zasadę Maksimum istotnie zależy od jakości trajektorii początkowej. Oznacza to, że należy oczekiwać trudności obliczeniowych, jeśli nie uda się znaleźć trajektorii początkowej, która jest bliska trajektorii optymalnej. Zastosowanie metody Newtona do zadań planowania ruchu robotów mobilnych zawierają prace [DW94, FGL94, LW95, Son93]. Metodę wyznaczenia rozwiązania fundamentalnego równania (9.15) (niezbędnego kroku w algorytmie Newtona dla robotów nieholonomicznych) zaczerpnięto z książki [Gan88]. Aspekty obliczeniowe metody Newtona badał Jakubiak [Jak99]. Divilbiss i Wen pokazali jak metodę Newtona można adaptować dla środowiska kolizyjnego. Autorzy ci zaproponowali, żeby każda przeszkoda wyznaczała zbiór zabronionych wartości  $\lambda$ , gdy trajektoria odpowiadająca sterowaniu (9.11) koliduje z przeszkodą. Metodą funkcji kary definiuje się funkcje (różniczkowalne) o wartościach 0 w przestrzeni bezkolizyjnej i o bardzo dużych wartościach wewnątrz przeszkód. Następnie warunek osiągnięcia celu jest rozszerzany — nie tylko ma być osiągnięty punkt docelowy, lecz także wartość funkcji kary ma wynosić 0\*. W przestrzeni rozszerzonej poszukiwania są prowadzone także metodą Newtona. Sformułowanie metody

---

\*Co automatycznie oznacza bezkolizyjność trajektorii.

uśredniania można znaleźć w monografii Arnolda, [Arn78]. Wykorzystanie metody uśredniania do planowania ruchu robotów nieholonomicznych zawierają prace Gurbits'a i współpracowników, [LC93, Gur92]. Podwaliny metody Lie-algebraicznej położył Hermes w pionierskich pracach o stabilizacji układów afinicznych, [Her78, Her80]. Idee Hermesa do zadań planowania ruchu robotów nieholonomicznych adaptował Dulęba, [Dul95, Dul97, Dul98a]. Techniczny, lecz bardzo ważny i równocześnie najtrudniejszy element metody Lie-algebraicznej to generacja sterowań realizujących zadany kierunek. Zastosowanie formuły Campbella-Bakera-Hausdorffa-Dynkina, dającej sterowania odcinkami stałe o dużej liczbie przełączeń, pochodzi od Hermesa [Her78]. Uogólnioną wersję tej formuły, pozwalającą na generowanie sterowań odcinkami ciągłych o mniejszej liczbie przełączeń, zaczerpnięto od Strichartz, [Str87]. Efektywne obliczeniowo zastosowanie formuły uogólnionej zawiera artykuł [Dul98b]. Istnieje cała rodzina metod dyskretnych, [BL93, BL89, FR96, Fer96], służących rozwiązaniu zadania planowania ruchu robotów nieholonomicznych. Metoda dyskretna polega na dyskretyzacji przestrzeni sterowań, czyli wybraniu jedynie kilku możliwych zestawów sterowań jako dopuszczalnych w danym punkcie przestrzeni stanu. Zwykle są to trzy wartości dla każdego sterowania: maksymalna, minimalna i neutralna (zerowa). Sterowania te, stosowane przez krótki czas w stanie bieżącym, generują zbiór stanów (rozwiniecie bieżącego stanu), który jest dodawany do zbioru stanów oczekujących na rozwinięcie. Jeden stan z tego zbioru staje się nowym stanem bieżącym i do niego rekurencyjnie stosowana jest procedura rozwijania. Wybór nowego stanu bieżącego reguluje procedura ukierunkowująca przeszukiwania tak, by rozwijane były stany mające szansę osiągnięcia stanu docelowego. Zwykle jest to pewien wariant algorytmu  $A^*$ , [Pod96]. Metoda dyskretna jest zatem techniką przeszukiwania grafu stosowaną do dyskretnego modelu robota nieholonomicznego. Jest ona zbieżna, o ile jednokrokový czas sterowania jest krótki. Do tego nie jest wrażliwa na istnienie przeszkód w otoczeniu robota (wygenerowane stany kolizyjne z przeszkodami są wykluczane), ale niestety bardzo złożona obliczeniowo. Krótki czas sterowania jednokrokowego, gwarantujący zbieżność metody, powoduje, że ruch elementarny jest krótki, a zatem liczba kroków potrzebna do osiągnięcia celu znaczna. Innym sposobem przeszukiwania grafu stanu jest metoda propagacji fali. Fala zainicjowana w stanie początkowym jest propagowana na zasadzie przyrostu kosztu ruchu ocenianego pewną funkcją kryterialną. Czoło fali pokazuje stany jednakowo odległe w sensie kosztu od stanu początkowego. Gdy osiągnie ono stan doce-

lowy, rozwiązanie zadania planowania ruchu jest znalezione. Metodę tę stosuje się od dawna, np. przy projektowaniu ścieżek układów elektronicznych. Istnieją liczne warianty metody propagacji fali, w których np. fala jest inicjowana w punkcie docelowym, [LP96]. Podstawowym problemem przy efektywnym stosowaniu metod przeszukiwania grafu jest umiejętne określenie kosztu elementarnego przemieszczenia. Zwykle stosuje się reguły zdroworozsądkowe, opisujące koszt przemieszczenia jako pewną ważoną funkcję przeorientowania kąтового i przestrzennego robota mobilnego. Dla najprostszych robotów ta heurystyka jest efektywna, dla robotów o wielowymiarowej przestrzeni stanu (np. ciągnących przyczepy) intuicja euklidesowa używana przy konstrukcji funkcji kosztu dla przeszukiwania grafu może zawieść. Należy wtedy uciec się do aparatu matematycznego opisującego nieholonomiczne sfery geodezyjne, których kształt dalece odbiega od euklidesowych, [SD98, VG94]. Kolejnym sposobem planowania ruchu robotów nieholonomicznych jest wykorzystanie metody pól potencjałowych, [Ste89]. Metoda ta jest chętnie wykorzystywana do rozwiązywania różnych zadań robotycznych ze względu na jej ogólność. Istota metody pól potencjałowych polega na określeniu w bieżącym punkcie przestrzeni stanu wektorowej reprezentacji wpływu stanu docelowego (siła przyciągająca) i przeszkód (siły odpychające). Gra przeciwstawnych sił generuje lokalnie kierunek ruchu. Wadą tej metody jest możliwość zerowania wektora sił wypadkowych w punkcie, który nie jest stanem docelowym. Aby wyprowadzić algorytm z punktu stacjonarnego należy stosować specjalne metody (np. generacji lokalnego podcelu). Cechy metod ogólnego przeznaczenia posiada metoda probabilistyczna planowania ruchu, [SO98]. Metoda ta jest zwykle stosowana w środowisku kolizyjnym i składa się z trzech etapów. Najpierw budowana jest mapa (graf) konfiguracji (wierzchołków) przez losowe wybieranie konfiguracji bezkolizyjnych i łączenie ich trajektoriami dopuszczalnymi — gałęziami grafu, z zastosowaniem lokalnego układu planowania trajektorii. W fazie drugiej, dla zadanego punktu początkowego i końcowego, poszukiwane są trajektorie dopuszczalne łączące te punkty z punktami wygenerowanymi w etapie pierwszym (ponownie wykorzystywany jest lokalny układ planowania trajektorii). W ostatnim etapie prowadzi się poszukiwanie trajektorii biegnących przez wierzchołki grafu od wierzchołka odpowiadającego punktowi początkowemu do wierzchołka odpowiadającego punktowi końcowemu. Idea metody probabilistycznej przypomina znaną dla holonomicznych robotów mobilnych technikę diagramów Woronia, [Lat93]. Środowisko kolizyjne stawia przed metodami planowa-

nia ruchu wysokie wymagania obliczeniowe, a obliczeniowo efektywne zastosowanie precyzyjnych metod analitycznych, sprawdzonych w środowisku bezkolizyjnym, do nieuchronnie kolizyjnych środowisk rzeczywistych ciągle oczekuje na realizację. Odrębnym, związanym z tym problemem, jest wierne modelowanie środowiska kolizyjnego dla nawigacji i lokalizacji robota, z wykorzystaniem sensorów i ich układów. W tej dziedzinie sporo osiągnięć mają młodzi polscy naukowcy, których dorobek, wraz z obszerną literaturą przedmiotu, zawierają rozprawy doktorskie [Dra99, Dub99, Kre97, Pia94, Rac95, Sie96, Skr97]. Głównym źródłem informacji o otoczeniu robota najczęściej są w tych pracach sonary ultradźwiękowe ze względu na dużą ilość informacji o otoczeniu przez nie dostarczanych oraz niskie koszty zakupu i eksploatacji. Metody analityczne nadają się do planowania trajektorii ruchu przy założeniu jednorodności powierzchni, po której porusza się robot. Obecnie to założenie jest uzasadnione, gdyż roboty mobilne pracują głównie w środowiskach specjalnie przygotowanych (płaska, jednorodna podłoga o odpowiednim współczynniku tarcia). Jednak wraz z ekspansją robotów mobilnych do środowisk otwartych oczekujemy wzrostu znaczenia metod lokalnych, uwzględniających zmienność środowiska. Zmienne w czasie (dynamiczne) środowisko wymusi także wzrost zainteresowania modyfikacjami modeli (i to prawdopodobnie raczej strukturalnymi niż parametrycznymi), gdy istotne staną się interakcje kół z podłożem.

## Literatura

- [AF69] M. Athans i P. L. Falb, *Sterowanie optymalne. Wstęp do teorii i jej zastosowania*. WNT, Warszawa, 1969.
- [Arn78] V. I. Arnold, *Mathematical Methods of Classical Mechanics*. Springer-Verlag, Berlin, 1978.
- [BL89] J. Barraquand i J. C. Latombe, On non-holonomic mobile robots and optimal maneuvering. *Revue d'Intelligence Artificielle*, **3**(2):77–103, 1989.
- [BL93] J. Barraquand i J. C. Latombe, Nonholonomic multibody mobile robots: Controllability and motion planning in the presence of obstacles. *Algorithmica*, **10**:121–155, 1993.
- [Dra99] P. Drapikowski, *Trójwymiarowe modelowanie otoczenia robota mobilnego*. Rozprawa doktorska, Politechnika Poznańska, Poznań, 1999.

- [Dub99] A. Dubrawski, *Neural networks for self-localization of mobile robots*. Rozprawa doktorska, Instytut Podstawowych Problemów Techniki PAN, Warszawa, 1999.
- [Dul95] I. Dułęba, Planowanie trajektorii układów nieholonomicznych na bazie algorytmu Hermesa. Raport nr 45, Instytut Cybernetyki Technicznej, Politechnika Wrocławska, 1995.
- [Dul97] I. Dułęba, Locally optimal motion planning of nonholonomic systems. *J. Robotic Systems*, **14**(11):767–788, 1997.
- [Dul98a] I. Dułęba, *Algorithms of Motion Planning for Nonholonomic Robots*. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 1998.
- [Dul98b] I. Dułęba, On a computationally simple form of the generalizad Campbell-Baker-Hausdorff-Dynkin formula. *Systems & Contr. Lett.*, **34**:191–202, 1998.
- [DW94] A. W. Divilbiss i J. T. Wen, Nonholonomic path planning with inequality constraints. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. **1**, strony 52–57, San Diego, 1994.
- [Fer96] P. Ferbach, A method of progressive constraints for nonholonomic motion planning. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. **3**, strony 2949–2955, Minneapolis, 1996.
- [FGL94] C. Fernandes, L. Gurvits i Z. X. Li, Near-optimal nonholonomic motion planning for a system of coupled rigid bodies. *IEEE Trans. Autom. Contr.*, **39**(3):450–463, 1994.
- [FR96] P. Ferbach i J. F. Rit, Planning nonholonomic motions for manipulated objects. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. **3**, strony 2935–2942, Minneapolis, 1996.
- [FSW80] W. Findeisen, J. Szymanowski i A. Wierzbicki, *Teoria i metody obliczeniowe optymalizacji*. PWN, Warszawa, 1980.
- [Gan88] F. R. Gantmacher, *Teoria matric*. Nauka, Moskwa, 1988.
- [Gur92] L. Gurvits, Averaging approach to nonholonomic motion planning. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. **3**, strony 2541–2545, Nice, 1992.
- [Her78] H. Hermes, Lie algebras of vector fields and local approximation of attainable sets. *SIAM J. Contr. Opt.*, **16**(6):715–728, 1978.
- [Her80] H. Hermes, On the synthesis of a stabilizing feedback control via Lie algebraic method. *SIAM J. Contr. Opt.*, **18**(6):352–361, 1980.

- [Jak99] J. Jakubiak, Numeryczne aspekty metody Newtona planowania ruchu robotów mobilnych. Praca magisterska, Instytut Cybernetyki Technicznej, Politechnika Wrocławska, 1999.
- [Kre97] B. Kreczmer, *Nawigacja robota mobilnego w środowisku quasi-statycznym*. Rozprawa doktorska, Politechnika Wrocławska, Wrocław, 1997.
- [Lat93] J. C. Latombe, *Robot Motion Planning*. Kluwer, Boston, 1993.
- [LC93] Z. Li i J. Canny, (red.), *Nonholonomic Motion Planning*. Kluwer, New York, 1993.
- [LP96] P. Lajmert i L. Podsędkowski, Global path planner for mobile robot with nonholonomic constraints. W: *Proc. MMAR Symposium*, vol. 3, strony 1023–1028, Międzyzdroje, 1996.
- [LW95] F. Lizarralde i J. T. Wen, Feedback stabilization of nonholonomic systems based on path space iteration. W: *Proc. MMAR Symposium*, vol. 2, strony 485–490, Międzyzdroje, 1995.
- [Pia94] M. Piasecki, *Metody globalnej lokalizacji robotów mobilnych na podstawie danych z sensorów niskiego poziomu*. Rozprawa doktorska, Politechnika Wrocławska, Wrocław, 1994.
- [Pod96] L. Podsędkowski, Generic model of multibody mobile robot for motion planning in the complex environment. W: *Proc. MMAR Symposium*, vol. 3, strony 1029–1034, Międzyzdroje, 1996.
- [Rac95] J. Racz, *Elementy sztucznej inteligencji w systemach nawigacyjnych autonomicznych mobilnych robotów*. Rozprawa doktorska, Instytut Podstawowych Problemów Techniki PAN, Warszawa, 1995.
- [SD98] J. Sówka i I. Dułęba, Sfery dla nieholonomicznych systemów bezdryfowych. W: *Materiały 5 Krajowej Konferencji Robotyki*, vol. 1, strony 161–169, Świeradów Zdrój, 1998.
- [Sie96] B. Siemiątkowska, *Rastrowa reprezentacja otoczenia w sterowaniu ruchomym robotem*. Rozprawa doktorska, Instytut Podstawowych Problemów Techniki PAN, Warszawa, 1996.
- [Skr97] P. Skrzypczyński, *Budowa geometrycznych map otoczenia dla celów nawigacji autonomicznego robota mobilnego*. Rozprawa doktorska, Politechnika Poznańska, Poznań, 1997.
- [SO98] P. Svestka i M. Overmars, Probabilistic path planning. W: J. P. Laumond, (red.), *Robot Motion Planning and Control*, vol. 229 serii *Lecture Notes in Control and Information Sciences*. Springer-Verlag, New York, 1998.

- [Son93] E. Sontag, Gradient technique for systems with no drift: A classical idea revisited. W: *Proc. IEEE Conf. Decision Control*, vol. **3**, strony 2706–2711, San Antonio, 1993.
- [Ste89] B. Steer, Trajectory planning for a mobile robot. *Int. J. Robotics Research*, **8**(5):3–14, 1989.
- [Str87] R. S. Strichartz, The Campbell-Baker-Hausdorff-Dynkin formula and solutions of differential equations. *J. Funct. Analysis*, **72**:320–345, 1987.
- [VG94] A. M. Vershik i V. Ya. Gershkovich, Nonholonomic dynamical systems, geometry of distribution and variational problems. W: V. I. Arnold, (red.), *Dynamical Systems VII*, strony 1–81. Springer-Verlag, Berlin, 1994.
- [VG97] T. L. Vincent i W. J. Grantham, *Nonlinear and Optimal Control Systems*. J. Wiley and Sons, New York, 1997.





## Rozdział 10

# Specjalizowane metody planowania ruchu

Specjalizowane metody planowania ruchu robotów nieholonomicznych mogą być zastosowane tylko do tych robotów, których model matematyczny spełnia, oprócz sterowalności, dodatkowe warunki regularności.

W tym rozdziale objaśnimy na prostych przykładach działanie następujących metod specjalizowanych:

- Metody planowania optymalnych trajektorii ruchu jednokołowego robota mobilnego. Metody te bazują na Zasadzie Maksimum Pontriagina, jednak są zaliczane do metod specjalizowanych ze względu na wykorzystywanie szczególnych własności modelu.
- Metody sterowań sinusoidalnych stosowalnej do robotów nieholonomicznych, których modele dadzą się sprowadzić do postaci łańcuchowej.
- Metody planowania ruchu koła toczącego się po płaskiej powierzchni wykorzystującą twierdzenie Stokesa.
- Metody kreacji i realizacji podcelów w planowaniu ruchu jednokołowego robota mobilnego.

### 10.1 Optymalne planowanie ruchu jednokołowego robota mobilnego

Specyficzne zastosowanie Zasady Maksimum Pontriagina ma miejsce przy konstrukcji czaso-optymalnych trajektorii jednokołowego robota mobilnego

o wektorze stanu  $\mathbf{q} = (x, y, \theta)^T$ , opisanego równaniami

$$\dot{\mathbf{q}} = \mathbf{g}_1(\mathbf{q})u_1 + \mathbf{g}_2(\mathbf{q})u_2, \quad (10.1)$$

gdzie  $\mathbf{g}_1(\mathbf{q}) = (\cos \theta, \sin \theta, 0)^T$ ,  $\mathbf{g}_2(\mathbf{q}) = (0, 0, 1)^T$  z dwoma rodzajami ograniczeń na prędkość liniową  $u_1$  oraz na prędkość kątową  $u_2$  zadanych warunkami:

- $u_1 = \pm 1$ ,  $u_2 \in [-1, 1]$  definiującymi zadanie *Reedsa-Sheppa*, w którym robot może poruszać się w przód i w tył,
- $u_1 = 1$ ,  $u_2 \in [-1, 1]$  definiującymi zadanie *Dubinsa*, w którym robot może poruszać się jedynie do przodu.

Zadanie planowania ruchu jednokołowego robota mobilnego jest jednym z nielicznych przykładów zadania planowania, które można rozwiązać analitycznie. Sposób rozwiązania tego zadania polega na konstrukcji rodzin ekstremal, które spełniają równania kanoniczne Hamiltona. Spośród wyznaczonych ekstremal wybierane jest rozwiązanie optymalne. Niestety, pomimo że ogólne zasady analizy ekstremal mogą zostać zastosowane do wszystkich rodzajów problemów planowania ruchu, to jednak tylko dla najprostszych układów nieholonomicznych ekstremale daje się scharakteryzować analitycznie.

W niniejszym podrozdziale pokażemy zasady konstrukcji sparametryzowanych rodzin ekstremal dla zadania Reedsa-Sheppa, wśród których znajduje się trajektoria optymalna. Omówimy sposób syntezy trajektorii optymalnej łączącej punkty  $(x, y, \theta)^T$  i  $(0, 0, 0)^T$  przestrzeni stanu, naszkicujemy sposób znajdowania trajektorii optymalnych dla zadania Dubinsa, oraz w sposób syntetyczny przedstawimy algorytm planowania ruchu dla jednokołowego robota mobilnego wykorzystujący Zasadę Maksimum Pontriagina.

### 10.1.1 Rodziny ekstremal dla zadania Reedsa-Sheppa

Dla ułatwienia zastosowania Zasady Maksimum Pontriagina rozważmy problem nieco ogólniejszy od przedstawionego powyżej, w którym także sterowania  $u_1$  należą do przedziału  $[-1, 1]$ . Rozwiązanie zadania Reedsa-Sheppa powstaje jako specyfikacja rozwiązania zadania uogólnionego.

Funkcja Hamiltona dla zadania czaso-optymalnego sterowania jednokołowego robota mobilnego jest zadana formułą\*

$$H(\mathbf{q}, \mathbf{p}, p_0, \mathbf{u}) = -p_0 \cdot 1 + \langle \mathbf{p}, \dot{\mathbf{q}} \rangle = -p_0 + p_1 \cos \theta u_1 + p_2 \sin \theta u_1 + p_3 u_2 = -p_0 + \phi_1 u_1 + \phi_2 u_2, \quad (10.2)$$

gdzie  $p_0$  jest stałą,  $\mathbf{p} \in \mathbb{R}^3$  jest zmienną dołączoną, a  $\phi_1 = \langle \mathbf{p}, \mathbf{g}_1 \rangle$ ,  $\phi_2 = \langle \mathbf{p}, \mathbf{g}_2 \rangle$  są funkcjami przełączającymi. Równania kanoniczne Hamiltona dla zmiennych dołączonych są następujące

$$\begin{cases} \dot{p}_1 = -\frac{\partial H}{\partial x} = 0 \\ \dot{p}_2 = -\frac{\partial H}{\partial y} = 0 \\ \dot{p}_3 = -\frac{\partial H}{\partial \theta} = p_1 \sin \theta u_1 - p_2 \cos \theta u_1 = p_1 \dot{y} - p_2 \dot{x}. \end{cases} \quad (10.3)$$

Zasada Maksimum Pontriagina stanowi warunek konieczny optymalności sterowań  $u_1$ ,  $u_2$  dla zadania z minimalizacją czasu sterowania

$$p_0 = \langle \mathbf{p}, \mathbf{g}_1(\mathbf{q})u_1 + \mathbf{g}_2(\mathbf{q})u_2 \rangle = \max_{(v_1, v_2) \in [-1, 1] \times [-1, 1]} (\langle \mathbf{p}, \mathbf{g}_1(\mathbf{q}) \rangle v_1 + \langle \mathbf{p}, \mathbf{g}_2(\mathbf{q}) \rangle v_2). \quad (10.4)$$

Warunek (10.4) wiedzie do sterowań

$$\begin{cases} u_1(t) = \operatorname{sgn} \phi_1(t) \\ u_2(t) = \operatorname{sgn} \phi_2(t). \end{cases} \quad (10.5)$$

Wyliczmy pochodne funkcji przełączających wykorzystując wzór (2.154)

$$\dot{\phi}_i = \frac{d}{dt} \mathbf{p}^T \mathbf{g}_i(\mathbf{q}) = \mathbf{p}^T \sum_{j=1}^m [\mathbf{g}_j, \mathbf{g}_i](\mathbf{q}) u_j = \langle \mathbf{p}, [\mathbf{g}_1, \mathbf{g}_i] u_1 + [\mathbf{g}_2, \mathbf{g}_i] u_2 \rangle.$$

Podstawiając za  $i = 1, 2$  i wykorzystując własność  $[\mathbf{g}_i, \mathbf{g}_i] = \mathbf{0}$  otrzymujemy

$$\begin{cases} \dot{\phi}_1 = u_2 \langle \mathbf{p}, [\mathbf{g}_2, \mathbf{g}_1] \rangle \\ \dot{\phi}_2 = -u_1 \langle \mathbf{p}, [\mathbf{g}_2, \mathbf{g}_1] \rangle. \end{cases}$$

Zdefiniujmy nową funkcję  $\phi_3 = \langle \mathbf{p}, [\mathbf{g}_2, \mathbf{g}_1] \rangle$  skonstruowaną analogicznie do funkcji przełączających i nazwijmy ją, nieco nieformalnie, także funkcją

\*Zobacz dodatek A.5.

przełączającą. Stosując podobny sposób wyprowadzenia jak powyżej, łatwo wyliczyć pochodną funkcji  $\phi_3$ . Łącząc pochodne funkcji przełączających uzyskujemy następujące równania różniczkowe

$$\begin{cases} \dot{\phi}_1 = u_2 \phi_3 \\ \dot{\phi}_2 = -u_1 \phi_3 \\ \dot{\phi}_3 = -u_2 \phi_1. \end{cases} \quad (10.6)$$

Z Zasady Maksimum Pontriagina, po uwzględnieniu sterowań (10.5), uzyskuje się natychmiast równanie

$$|\phi_1(t)| + |\phi_2(t)| = p_0, \quad (10.7)$$

prawdziwe dla każdego  $t$  z przedziału  $[0, T]$ , gdzie  $T$  jest nieokreślonym czasem planowania ruchu robota. Ponieważ  $\mathbf{p}(t) \neq \mathbf{0}$ , oraz pola  $\mathbf{g}_1, \mathbf{g}_2, [\mathbf{g}_2, \mathbf{g}_1]$  rozpinają przestrzeń stanu w każdym punkcie, to w każdej chwili czasu przynajmniej jedna z funkcji  $\phi_i$ ,  $i = 1, 2, 3$  jest różna od zera

$$|\phi_1(t)| + |\phi_2(t)| + |\phi_3(t)| \neq 0. \quad (10.8)$$

Równania (10.5), (10.6), (10.7), (10.8), posłużą nam do sformułowania kilku interesujących własności ekstremal ( $\mathbf{u}(t), \mathbf{q}(t), \mathbf{p}(t)$ ) zadania sterowania czasooptymalnego, wymienionych poniżej.

1. Z równości (10.6) i (10.8) wnioskujemy, że nie istnieją niezerowe ekstremale osobliwe. Stąd wynika, że istnieją wyłącznie ekstremale regularne co znaczy, że możemy przyjąć  $p_0 = 1$ .
2. Z równania (10.8) wynika, że dla ekstremal regularnych suma modułów funkcji przełączających jest dodatnia, co dowodzi, że na ekstremalach regularnych  $\phi_1, \phi_2$  nie zerują się jednocześnie.
3. Z zależności (10.6) możemy wywnioskować, że wzdłuż wszystkich ekstremal  $\kappa = \phi_1^2 + \phi_3^2 = const$ . Pokażemy, że  $\kappa = 0 \Leftrightarrow \phi_1 \equiv 0$ . Implikacja  $\kappa = 0 \Rightarrow \phi_1 \equiv 0$  jest trywialna, zatem rozważymy implikację przeciwną. Dla  $\phi_1 \equiv 0$  z własności 2 mamy w każdej chwili  $\phi_2(t) \neq 0$ , a zatem  $u_2(t) \neq 0$ . Z trzeciego równania (10.6) wynika, że  $\phi_3 = const$ , natomiast pierwsze równanie (10.6) może być spełnione jedynie gdy  $\phi_3 = 0$ , co kończy dowód.
4. Wzdłuż ekstremali zera funkcji przełączającej  $\phi_1(t)$  są albo izolowane (i w nich  $\dot{\phi}_1 \neq 0$ ), albo funkcja ta zeruje się tożsamościowo na

przedziale czasu  $[0, T]$ . Drugą część stwierdzenia udowodniono we własności 3. Dla wykazania, że zera  $\phi_1$  są izolowane (co oznacza, że  $\phi_1(t) = 0 \Rightarrow \dot{\phi}_1 \neq 0$ ) założymy, że na ekstremali  $\phi_1(t) \neq 0$ , a zatem na pewnym przedziale zachodzi  $\kappa > 0$ . Gdyby  $\phi_1(t) = 0$  i  $\dot{\phi}_1(t) = 0$ , to  $\phi_2(t) \neq 0$  na pewnym przedziale zawierającym  $t$ . Mamy  $u_2(t) \neq 0$ , a z (10.6) wynika  $\phi_3(t) \equiv 0$ . Stąd  $\kappa = 0$ , a więc  $\phi_1(t) \equiv 0$ .

Z ostatniej własności ekstremal wynikają dwa możliwe typy trajektorii:

**typ A** charakteryzujący się skończoną liczbą przełączeń sterowania  $u_1$ .

**typ B** opisujący trajektorie, dla których  $\phi_1(t) \equiv 0$  i albo  $u_2 \equiv 1$ , albo  $u_2 \equiv -1$ .

Aby pokazać technikę konstrukcji ekstremal potencjalnie optymalnych przeanalizujemy szczegółowo przypadek A. Rozważymy najpierw ekstremale odpowiadające sterowaniu  $u_1 = 1$  na przedziale czasu  $[0, T]$ <sup>†</sup>. Funkcja przełączająca  $\phi_1(t)$  nie zeruje się zatem w tym przypadku na przedziale otwartym  $(0, T)$  (i jest w nim dodatnia). Drugie i trzecie z równań (10.6), dla sterowania  $u_1 = 1$ , są następujące:

$$\begin{cases} \dot{\phi}_2 = -\phi_3 \\ \dot{\phi}_3 = -u_2\phi_1. \end{cases} \quad (10.9)$$

Podstawiając optymalne sterowanie  $u_2$  i zastępując dwa równania (10.9) jednym równaniem drugiego rzędu otrzymujemy

$$\ddot{\phi}_2 = \phi_1 \operatorname{sgn}(\phi_2),$$

co oznacza, że  $\phi_2$  jest funkcją wypukłą (wklęsłą) na przedziale czasu  $[0, T]$ , gdy  $\phi_2 > 0$  ( $\phi_2 < 0$ ).

Niech  $\mathbb{C}_a$  opisuje ruch po łuku okręgu o mierze kątowej  $a$ , a  $\mathbb{S}_c$  oznacza ruch wzdłuż segmentu prostoliniowego o długości  $c$ . Wówczas trajektorie potencjalnie optymalne dla sytuacji A ze sterowaniem  $u_1 = 1$  należą do jednego z trzech rodzajów:

1.  $\mathbb{C}_a$ ,  $0 \leq a \leq \pi$ .

Sterowania  $u_1 = 1$ ,  $u_2 = \pm 1$ ,  $t \in [0, T]$ . Taki scenariusz odpowiada warunkowi  $\phi_2(t) \neq 0$  dla  $t \in [0, T]$ . Trajektorja zainicjowana w punk-

<sup>†</sup>Analogicznie, przez symetrię, rozwiązuje się przypadek  $u_1 = -1$ .

cie  $(x_0, y_0, \theta_0)^T$  dla sterowania  $u_2 = \pm 1$  jest następująca:

$$\begin{cases} x(t) = x_0 \pm (\sin(\theta_0 \pm t) - \sin(\theta_0)) \\ y(t) = y_0 \mp (\cos(\theta_0 \pm t) - \cos(\theta_0)) \\ \theta(t) = \theta_0 \pm t, \end{cases} \quad t \in [0, T].$$

Powyższe równania przedstawiają łuk okręgu o promieniu 1 i środku w punkcie  $(x_0 \mp \sin(\theta_0), y_0 \pm \cos(\theta_0))^T$ . Wartość parametru  $a$  nie może być większa od  $\pi$ , gdyż poruszając się po łuku okręgu w przeciwnym kierunku uzyskuje się krótszą drogę kątową  $2\pi - a$ .

2.  $\mathbb{C}_a \mathbb{S}_c \mathbb{C}_b$ ,  $0 \leq a \leq \pi/2$ ,  $0 < c$ ,  $0 \leq b \leq \pi/2$ .

Gdy  $\phi_2(t)$  zeruje się na pewnym podprzedziale przedziału czasu  $[0, T]$ , to przedział  $[0, T]$  można podzielić na trzy podprzedziały  $\mathcal{I}_1 = [0, t']$ ,  $\mathcal{I}_2 = [t', t'']$ ,  $\mathcal{I}_3 = [t'', T]$ . Na sumie  $\mathcal{I}_1 \cup \mathcal{I}_3$  funkcja przełączająca  $\phi_2(t)$  nie zeruje się, natomiast na  $\mathcal{I}_2$  zeruje się ona tożsamościowo. Zatem, na przedziałach  $\mathcal{I}_1, \mathcal{I}_3$  sterowanie  $u_2 = \pm 1$ , natomiast na przedziale  $\mathcal{I}_2$  powinno ono należeć do przedziału  $[-1, 1]$ . Dalej pokażemy, że sterowanie  $u_2$  na przedziale  $\mathcal{I}_2$  jest równe zero.

Wykorzystując własność wklęsłości (wypukłości) funkcji  $\phi_2(t)$  na przedziale  $[0, t']$  łatwo pokazać, korzystając z równań (10.6), że funkcja  $\phi_3(t)$  i jej pochodna  $\dot{\phi}_3(t)$  mają stały znak na przedziale  $(0, t')$ . By pokazać dlaczego górnym ograniczeniem na parametry  $a, b$  jest  $\pi/2$  skorzystajmy z równań dołączonych. Własnością modelu jednokołowego robota mobilnego jest łatwość całkowania równań dołączonych (10.3), z których wynika, że  $p_1(t), p_2(t)$  są stałe w przedziale czasu  $[0, T]$  na trajektoriach optymalnych. Korzystając z definicji funkcji  $\phi_3(t) = \langle \mathbf{p}, [\mathbf{g}_2, \mathbf{g}_1] \rangle$ , oraz wyliczając nawias Liego  $[\mathbf{g}_2, \mathbf{g}_1] = (-\sin \theta, \cos \theta, 0)^T$  uzyskujemy wyrażenia

$$\begin{aligned} \phi_3(t) &= -p_1 \sin \theta + p_2 \cos \theta, \\ \dot{\phi}_3(t) &= -p_1 \cos \theta - p_2 \sin \theta. \end{aligned} \quad (10.10)$$

Równania (10.10) można zapisać jako

$$\begin{aligned} \phi_3(t) &= \sqrt{p_1^2 + p_2^2} \sin(\alpha + \theta), \\ \dot{\phi}_3(t) &= \sqrt{p_1^2 + p_2^2} \cos(\alpha + \theta), \end{aligned}$$

gdzie  $\cos \alpha = -\frac{p_1}{\sqrt{p_1^2 + p_2^2}}$ ,  $\sin \alpha = \frac{p_2}{\sqrt{p_1^2 + p_2^2}}$ . Korzystając z argumentu

o stałym znaku  $\phi_3(t)$ ,  $\dot{\phi}_3(t)$  na przedziale  $(0, t')$  oraz uwzględniając, że  $u_2 = 1$  lub  $-1$  na tym przedziale<sup>‡</sup> wnioskujemy, że długość przedziału  $(0, t')$  nie może być większa od  $\pi/2$ . Teraz pokażemy, dlaczego na przedziale  $[t', t'']$  sterowanie osobiwe  $u_2$  ( $\phi_2(t) \equiv 0$ ) ma wartość równą 0. Z równania (10.2) dla  $u_1 = 1$  uzyskuje się warunek

$$p_1 \cos \theta + p_2 \sin \theta = p_0. \quad (10.11)$$

Stałe  $p_1$ ,  $p_2$  nie mogą być jednocześnie zerami, gdyż wtedy  $\phi_1 = 0$  (zobacz (10.2)). Zatem, by równanie (10.11) mogło być spełnione w otwartym przedziale,  $\theta$  musi być stałe, co oznacza, że sterowanie  $u_2$  jest w tym przedziale zerowe.

Zauważmy, że omawiany scenariusz jest realizowany na przykład następującą sekwencją sterowań:  $(+1, +1)$ ,  $(+1, 0)$ ,  $(+1, -1)$ . Dla tej sekwencji wyliczamy trajektorię zainicjowaną w punkcie  $(x_0, y_0, \theta_0)^T$ . W przedziale czasu  $[0, a)$  trajektoria jest opisana zależnościami

$$\begin{cases} x(t) = x_0 + \sin(\theta_0 + t) - \sin \theta_0 \\ y(t) = y_0 - \cos(\theta_0 + t) + \cos \theta_0 \\ \theta(t) = \theta_0 + t. \end{cases}$$

Dla  $t \in (a, a + c)$  trajektoria jest prostoliniowa

$$\begin{cases} x(t) = x_0 + \sin(\theta_0 + a) - \sin \theta_0 + \cos(\theta_0 + a)(t - a) \\ y(t) = y_0 - \cos(\theta_0 + a) + \cos \theta_0 + \sin(\theta_0 + a)(t - a) \\ \theta(t) = \theta_0 + a, \end{cases}$$

a na końcowym segmencie, dla  $t \in (a + c, a + c + b]$  — łukiem okręgu

$$\begin{cases} x(t) = x_0'' - \left( \sin(\theta_0'' - t + a + c) - \sin \theta_0'' \right) \\ y(t) = y_0'' + \left( \cos(\theta_0'' - t + a + c) - \cos \theta_0'' \right) \\ \theta(t) = \theta_0'' - t + a + c, \end{cases}$$

gdzie

$$\begin{cases} x_0'' = x_0 + \sin(\theta_0 + a) - \sin \theta_0 + \cos(\theta_0 + a)c \\ y_0'' = y_0 - \cos(\theta_0 + a) + \cos \theta_0 + \sin(\theta_0 + a)c \\ \theta_0'' = \theta_0 + a. \end{cases}$$

<sup>‡</sup>Co wiedzie do ciągłego przeorientowywania robota:  $\theta(t) = \theta_0 \pm t$ .

3.  $C_a C_b$ ,  $0 \leq a \leq \pi/2$ ,  $0 \leq b \leq \pi/2$ .

Ten scenariusz jest szczególnym przypadkiem scenariusza poprzedniego, gdy segment prostoliniowy dąży do punktu,  $c = 0$ .

Jeszcze raz skorzystajmy z równań dołączonych, by pokazać na jakich krzywych są możliwe przełączenia sterowań. Niech  $x(t_0) = y(t_0) = 0$ . Z definicji funkcji przełączającej  $\phi_2 = \langle \mathbf{p}, \mathbf{g}_2 \rangle$  wynika równość  $\phi_2(t) = p_3(t)$ , natomiast z równania (10.3) uzyskujemy równanie łączące funkcję przełączającą ze zmiennymi stanu i zmiennymi dołączonymi

$$\phi_2(t) = p_3(t) = p_3(t_0) + p_1 y(t) - p_2 x(t). \quad (10.12)$$

Pełna charakterystyka możliwych punktów przełączeń sterowań dla trajektorii typu A jest następująca:

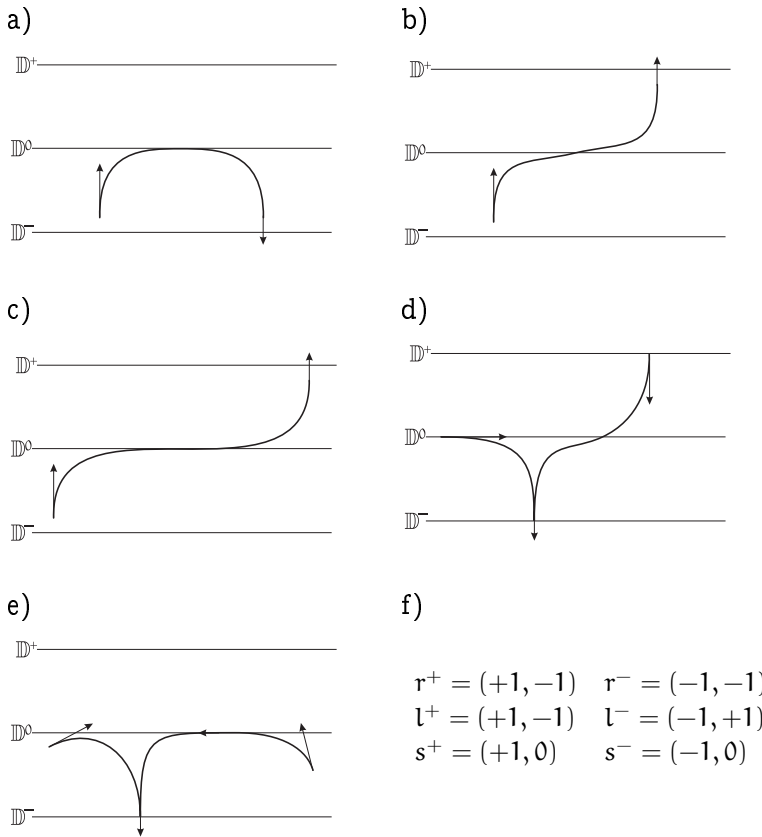
- jeżeli  $\phi_2(t) = 0$ , to punkt przełączenia sterowania  $u_2$  leży na prostej  $\mathbb{D}^0$ :  $p_1 y - p_2 x + p_3(t_0) = 0$ ;
- jeżeli  $\phi_1(t) = 0$ , wówczas z warunku koniecznego (10.4) wynika zależność  $p_3(t)u_2(t) - p_0 = 0$ , a z postaci (10.12) wniosek, że możliwe są dwa przypadki:
  - jeśli  $u_2 = 1$ , to punkty przełączeń leżą na prostej  $\mathbb{D}^+$  o równaniu  $p_1 y - p_2 x + p_3(t_0) - p_0 = 0$ ,
  - gdy  $u_2 = -1$ , wtedy punkty przełączeń leżą na prostej  $\mathbb{D}^-$  opisanej równaniem  $p_1 y - p_2 x + p_3(t_0) + p_0 = 0$ .

Proste  $\mathbb{D}^-$ ,  $\mathbb{D}^0$ ,  $\mathbb{D}^+$  są równoległe. Trajektoria optymalna może przebiegać wzdłuż prostej  $\mathbb{D}^0$  (segment prostoliniowy odpowiadający osobliwemu sterowaniu  $u_2$ ) lub być do niej styczna. Na prostych  $\mathbb{D}^+$ ,  $\mathbb{D}^-$  trajektoria optymalna zmienia kierunek (w jedynym punkcie styczności), co odpowiada zmianie znaku sterowania  $u_1$ . Podczas zmiany kierunku ruchu (w punkcie zwrotu), równanie (10.4) opisuje na płaszczyźnie  $\times Y$  prostą o równaniu  $\phi_1 = p_1 \cos \theta + p_2 \sin \theta = 0$ , prostopadłą do  $\mathbb{D}^-$ ,  $\mathbb{D}^0$ ,  $\mathbb{D}^+$ .

Geometryczną interpretację niektórych optymalnych trajektorii typu A przedstawiono na rysunku 10.1 ( $r/l$  oznacza, odpowiednio, skręt w prawo/lewo,  $s$  opisuje segment prostoliniowy, znak  $+/-$  określa kierunek ruchu, odpowiednio, do przodu i do tyłu).

Pełna lista rodzin trajektorii, wśród których znajduje się trajektoria optymalna, po uwzględnieniu trajektorii typu B, została podana w tabeli 10.1. Z tabeli tej wynika, że trajektoria optymalna dla zadania Reedsa-Sheppa nie





**Rysunek 10.1** Wybrane trajektorie optymalne typu A składające się z łuków okręgów i segmentów prostoliniowych: bez nawrotów: (a) typ  $r^+$ , (b) typ  $r^+l^+$ , (c) typ  $r^+s^+l^+$  oraz z nawrotami: (d) typ  $r^+r^-l^-$ , (e) typ  $r^+r_{\pi/2}^-s^-r^-$ , (f) typ ruchu i realizujące go sterowania  $(u_1, u_2)$ .

może mieć więcej niż dwa nawroty i pięć segmentów, w tym jeden segment prostoliniowy.

Dalej pokażemy jak znaleźć trajektorię optymalną łączącą zadany punkt przestrzeni stanu  $(x, y, \theta)^T$  z punktem  $(0, 0, 0)^T$ .

### 10.1.2 Synteza trajektorii optymalnej

Syntezę trajektorii optymalnych dla jednokołowego robota mobilnego przeprowadza się zastępując naturalną przestrzeń stanu  $SE(2) \cong \mathbb{R}^2 \times S^1$  ze stanem  $(x, y, \theta^*)^T$  przestrzenią  $\{x, y, \theta \mid \theta - \theta^* = 2k\pi, k \in \mathbb{Z}\} \subset \mathbb{R}^3$ , którą dzieli się płaszczyznami stałej wartości  $\theta$ . Na każdej z płaszczyzn  $\mathbb{P}_\theta$  wy-

rodzina	scenariusz ruchu <sup>†</sup>
(I)	$l_a^+ l_b^- l_e^+$ lub $r_a^+ r_b^- r_e^+$
(II) (III)	$C_a   C_b C_e$ lub $C_a C_b   C_e$
(IV)	$C_a C_b   C_b C_e$
(V)	$C_a   C_b C_b   C_e$
(VI)	$C_a   C_{\pi/2} S_l C_{\pi/2}   C_b$
(VII) (VIII)	$C_a   C_{\pi/2} S_l C_b$ lub $C_b S_l C_{\pi/2}   C_a$
(IX)	$C_a S_l C_b$
rodzina	parametry rodziny
(I)	$0 \leq a \leq \pi, \quad 0 \leq e \leq \pi, \quad 0 \leq b \leq \pi$
(II) (III)	$0 \leq a \leq b, \quad 0 \leq e \leq b, \quad 0 \leq b \leq \pi/2$
(IV)	$0 \leq a < b, \quad 0 \leq e < b, \quad 0 \leq b \leq \pi/2$
(V)	$0 \leq a < b, \quad 0 \leq e < b, \quad 0 \leq b \leq \pi/2$
(VI)	$0 \leq a < \pi/2, \quad 0 \leq b < \pi/2, \quad 0 \leq l$
(VII) (VIII)	$0 \leq a \leq \pi, \quad 0 \leq b \leq \pi/2, \quad 0 \leq l$
(IX)	$0 \leq a \leq \pi/2, \quad 0 \leq b \leq \pi/2, \quad 0 \leq l$

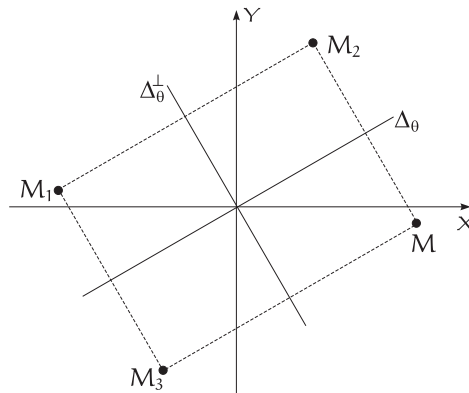
<sup>†</sup>r, l precyzuje ruch po łuku okręgu C: l — skręt w lewo, r — skręt w prawo, indeks górny wskazuje kierunek ruchu: + — do przodu, — — do tyłu, | oznacza nawrót, czyli zmianę kierunku ruchu.

**Tabela 10.1** Rodziny krzywych, wśród których znajduje się trajektoria optymalna.

kreślane są dwie proste: prosta  $\Delta_\theta$  o równaniu  $y = -x \operatorname{ctg}(\theta/2)$  i prostopadła do niej prosta  $\Delta_\theta^\perp$ , przechodząca przez początek układu współrzędnych, co pokazano na rysunku 10.2. Załóżmy, że znana jest optymalna trajektoria łącząca stan  $(M, \theta)^\top = (x, y, \theta)^\top$  ze stanem  $(0, 0, 0)^\top$ . Trajektoria ta jest zdeterminowana sekwencją sterowań optymalnych  $\mathbf{w}$  wyznaczających segmenty łuków okręgów i segmentów prostoliniowych. Zostało udowodnione, że trajektorie optymalne rozpoczynające się w punktach  $(M_i, \theta)$ ,  $i = 1, 2, 3$  (zobacz rysunek 10.2), symetrycznych względem prostych  $\Delta_\theta, \Delta_\theta^\perp$ , są realizowane sekwencjami sterowań  $\mathbf{w}_i$ , gdzie

- $\mathbf{w}_1$  jest sekwencją sterowań  $\mathbf{w}$  z zamienionymi górnymi indeksami + i —,
- $\mathbf{w}_2$  jest realizowane przez odwrócenie sekwencji  $\mathbf{w}$ , a następnie, zamienienie górnych indeksów + i —.
- $\mathbf{w}_3$  jest realizowane przez odwrócenie sekwencji  $\mathbf{w}$ .

Udało się wykazać symetrię trajektorii zainicjowanej w punkcie  $(M, \theta)$  i tra-



Rysunek 10.2 Podział płaszczyzny Reedsa-Sheppa.

rodzina	scenariusz ruchu
(I)	$l_a^+ l_b^- l_e^+$ lub $r_a^+ r_b^- r_e^+$
(VII) (VIII)	$C_a   C_{\pi/2} S_l C_b$ lub $C_b S_l C_{\pi/2}   C_a$
rodzina	parametry rodziny
(I)	$0 \leq a + b + e \leq \pi$
(VII) (VIII)	$0 \leq a \leq \pi/2, 0 \leq b \leq \pi/2, 0 \leq l$ jeśli $u_2$ jest stałe na wszystkich łukach to $a + b \leq \pi/2$

Tabela 10.2 Zmodyfikowane scenariusze ruchu dla niektórych rodzin trajektorii optymalnych zadania Reedsa-Sheppa, zobacz tabela 10.1.

jektorii przeprowadzającej stan  $(\bar{M}(x, -y), -\theta)$  w  $(0, 0, 0)$ . Sekwencja  $\bar{\mathbf{w}}$  powstaje przez zamianę w scenariuszu  $\mathbf{w}$  skrętu w prawo na skręt w lewo i odwrotnie. Ze względu na symetrię trajektorii optymalnych względem prostych  $\Delta_\theta, \Delta_\theta^\perp$  rozpatruje się jedynie jeden kwadrant płaszczyzny  $\mathbb{P}_\theta$ . Pokazano także, że scenariusze (I), (VII) i (VIII) zamieszczone w tabeli 10.1 mogą być zastąpione scenariuszami zawartymi w tabeli 10.2.

W ramach dalszych czynności przy wyznaczaniu optymalnej trajektorii ruchu jednokołowego robota mobilnego należy:

1. Dla każdej rodziny potencjalnie optymalnych trajektorii wyznaczyć na płaszczyźnie  $\mathbb{P}_\theta$ , ustalonej przez początkową orientację  $\theta$ , obszar odpowiadający wszystkim możliwym kombinacjom parametrów opisujących sekwencję sterującą zadaną dokładnie trzema parametrami  $a, e, b$  lub  $a, l, b$  (zobacz tabele 10.1, 10.2). Obszar ten powstaje nastę-

pująco:

- Całkujemy model (10.1) ze stanu początkowego  $(x, y, \theta)^T$  dla sterowań zadanych odpowiednią trójką parametrów. W ten sposób uzyskujemy zależne od parametrów trajektorie  $(x(t), y(t), \theta(t))^T$ . Z warunku końcowego  $(0, 0, 0)^T$  uzyskujemy układ trzech równań na trzy parametry.
  - Ponieważ płaszczyzna  $\mathbb{P}_\theta$  odpowiada ustalonej wartości początkowej kąta  $\theta$ , zatem tylko dwa parametry sekwencji sterującej są niezależne.
  - Przebiegając przedziały dopuszczalnych wartości tych dwóch parametrów uzyskujemy obszar na płaszczyźnie  $\mathbb{P}_\theta$ . W ten sposób każdej rodzinie trajektorii potencjalnie optymalnych przyporządkowany jest odpowiedni obszar na płaszczyźnie  $\mathbb{P}_\theta$ .
2. Ze względu na opisane uprzednio symetrie, obszary wyznaczone w punkcie 1 można ograniczyć do jednego wybranego kwadrantu. Gdy warunek początkowy  $(x, y, \theta)^T$  jest w innym kwadrancie niż wybrany, sprowadzamy warunek do wybranego kwadrantu korzystając z symetrii trajektorii względem prostych  $\Delta_\theta, \Delta_\theta^\perp$ .
  3. Jeżeli obszary wynikające z sekwencji sterowań potencjalnie optymalnych dla dwóch rodzin się przecinają, ustalić której rodziny trajektorie są optymalne na części wspólnej obszarów. Można to uczynić na drodze analitycznej lub numerycznej, przez wybranie dowolnego punktu wewnętrznego obszaru wspólnego i wyliczenie czasu sterowania dla każdej z potencjalnie optymalnych trajektorii. Rodzina, której sekwencja sterowań potencjalnie optymalnych generuje krótszy czas jest optymalna na części wspólnej obszarów.

Poniżej zamieszczamy przykład konstrukcji obszaru dla trajektorii  $l^+l^-l^+$ , należącej do rodziny (I).

### Przykład 10.1.1 (Konstrukcja trajektorii optymalnej)

Parametrami rodziny są  $a, b, e$  związane zależnością  $0 \leq a + b + e \leq \pi$ . Zakładamy, że  $\theta < 0$ . Jeśli jest inaczej, stan początkowy  $M$  transformujemy przez symetrię do  $\bar{M}$ , by nowy stan początkowy spełniał warunek ujemnego kąta  $\theta$ . Po wyliczeniu sterowań optymalnych dla nowego stanu początkowego należy je przetransformować zgodnie z regułami podanymi uprzednio, tj. zamienić skręt w lewo skrętem w prawo i odwrotnie,  $\mathbf{w} \rightarrow \bar{\mathbf{w}}$ .

scenariusz ruchu	parametry rodziny
$C_a S_l C_e$	$a \in [0, 2\pi), \quad e \in [0, 2\pi), \quad l \geq 0$
$C_a C_b C_e$	$\pi < b < 2\pi, \quad 0 \leq a \leq b, \quad 0 \leq e \leq b$ $0 \leq a < b - \pi \text{ lub } 0 \leq e < b - \pi$

**Tabela 10.3** Rodziny trajektorii optymalnych dla zadania Dubinsa (dopuszczalny ruch robota tylko do przodu,  $u_1 \equiv +1$ ).

Trajektorii  $l_a^+ l_b^- l_e^+$  odpowiada sekwencja sterowań  $(+1, +1)$ ,  $(-1, +1)$ ,  $(+1, +1)$ , włączonych, odpowiednio, przez czas  $a$ ,  $b$ ,  $e$ . Całkowanie równań modelu (10.1) dla tych sterowań z warunkami brzegowymi oraz punktami początkowym i końcowym trajektorii  $(x, y, \theta)^T$  i  $(0, 0, 0)^T$  prowadzi do równań

$$\begin{cases} x = \sin \theta + 2 \sin(b + e) - 2 \sin e \\ y = -\cos \theta + 2 \cos(b + e) - 2 \cos e + 1 \\ \theta = -a - b - e. \end{cases}$$

Ponieważ kąt  $\theta < 0$  jest ustalony, to parametr  $e \in [0, -\theta]$ , natomiast parametr  $b \in [0, -\theta - e]$ . Dla każdego ustalonego  $e$ , parametr  $b$  przebiegając swój zakres zmienności wyznacza łuk okręgu na płaszczyźnie  $X Y$  o promieniu 2 i środku w punkcie  $(\sin \theta - 2 \sin e, -\cos \theta - 2 \cos e + 1)^T$ . Łuki mają swe początki w punkcie  $(\sin \theta, -\cos \theta + 1)$  i w zależności od układu parametrów  $b$  i  $e$  różne długości. ■

Powyższy przykład pokazuje, że obszary optymalne dla rodzin istotnie zależą od płaszczyzny  $\mathbb{P}_\theta$ , na której są wyznaczone, czyli od orientacji początkowej  $\theta$ .

### 10.1.3 Planowanie ruchu jednokołowego robota mobilnego poruszającego się do przodu

Zadanie Dubinsa dla jednokołowego robota mobilnego polega na wyznaczeniu sterowań optymalnych przeprowadzających robota od zadanego stanu początkowego do zadanego stanu końcowego, gdy dopuszczalny jest wyłącznie ruch w przód. Sposób postępowania przy rozwiązywaniu zadania Dubinsa jest bardzo podobny do rozwiązywania problemu Reedsa-Sheppa. W zadaniu Dubinsa ruch planuje się od punktu  $(0, 0, 0)^T$  do punktu  $(x, y, \theta)^T$ . Trajektorii optymalnych należy poszukiwać wśród rodzin przedstawionych w tabeli 10.3.

Ponieważ liczba rodzin potencjalnie optymalnych jest istotnie mniejsza niż dla zadania Reedsa-Sheppa, zadanie Dubinsa w warstwie obliczeniowej jest znacznie łatwiejsze niż zadanie Reedsa-Sheppa. Różnica jest jednak ilościowa, a nie jakościowa. Robot mobilny poruszający się jedynie do przodu nie wykazuje tyłu symetrii co robot, dla którego jest także możliwy ruch do tyłu (symetrie wynikają głównie z symetrii dopuszczalnych sterowań). Jednak dla zadania Dubinsa istnieją także symetrie ze względu na prostą  $\Delta_{\theta}^{\perp}$ , oraz typu  $(M(x, y), \theta) \leftrightarrow (M(x, -y), -\theta)$ .

#### 10.1.4 Podsumowanie

Procedura wyznaczania rozwiązania zadania planowania ruchu jednokołowego robota mobilnego (z ograniczeniami na sterowania w wersji Reedsa-Sheppa) składa się z następujących kroków.

- Krok 1.** Przetransformować zadanie planowania ruchu od punktu początkowego  $(x_0, y_0, \theta_0)^T$  do punktu końcowego  $(x_d, y_d, \theta_d)^T$ , do zadania planowania ruchu od nowego punktu początkowego  $(x, y, \theta)^T = (x_a - x_d, y_0 - y_d, \theta_0 - \theta_d)^T$  do nowego punktu końcowego  $(0, 0, 0)^T$ .
- Krok 2.** Skonstruować rodziny krzywych spełniających warunek konieczny optymalności wynikający z Zasady Maksimum Pontriagina i dla każdej płaszczyzny  $\mathbb{P}_{\theta} = (x, y)^T$ , z ustalonym kątem  $\theta$  (parametr), określić obszary charakteryzujące optymalność trajektorii dla każdej z rodzin krzywych.
- Krok 3.** Określić, do której płaszczyzny  $\mathbb{P}_{\theta}$  i do którego obszaru na  $\mathbb{P}_{\theta}$  należy nowy punkt początkowy ruchu. W ten sposób ustala się (być może niejednoznacznie), jaka rodzina krzywych jest optymalna, czyli realizuje rozwiązanie zadania. Niejednoznaczność rozwiązania oznacza, że mogą istnieć dwie sekwencje sterowań (dwa rodzaje trajektorii) charakteryzujące się tym samym czasem realizacji trajektorii. Niejednoznaczność występuje, gdy zadany punkt leży na przecięciu obszarów optymalności dwóch rodzin trajektorii potencjalnie optymalnych.
- Krok 4.** Ustalona sekwencja sterowań optymalnych jest charakteryzowana zestawem trzech parametrów (kąty skrętu, długości segmentów prostoliniowych). Należy wyznaczyć taką wartość parametrów rodziny trajektorii optymalnych (przez sformułowanie odpowiednich równań wynikających z całkowania równania modelu dla wy-

branej rodziny trajektorii), by trajektoria rozpoczynająca się w nowym punkcie początkowym kończyła się w początku układu. Wyznaczone parametry charakteryzują jednoznacznie sekwencje sterowania realizujące trajektorię optymalną.

Pomimo, że zasady konstrukcji rozwiązania optymalnego wykorzystujące Zasadę Maksimum Pontriagina stosują się do każdego układu nieholonomicznego, bardzo trudno jest przy ich pomocy uzyskać analityczne rozwiązanie zadań planowania ruchu dla bardziej skomplikowanych modeli niż jednokołowy robot mobilny. Na czym zatem polega specyfika zadania planowania ruchu jednokołowego robota mobilnego? Okazuje się, że w jego przypadku istnieje liniowa zamiana współrzędnych, która transformuje zadanie planowania ruchu między dwoma dowolnymi punktami przestrzeni stanu, do zadania planowania między pewnym punktem przestrzeni a początkiem układu współrzędnych. W ten sposób rozwiązuje się „kanoniczny” przypadek planowania ruchu. Ponadto równania dołączone, ze względu na niezależność hamiltonianu od współrzędnych pozycyjnych  $(x, y)^T$ , są szczególnie łatwe do rozwiązania. W konsekwencji, zmienne dołączone  $p_1, p_2$  są stałe, co znacznie ułatwia badanie własności trajektorii optymalnych. I w końcu, ze względu na dopuszczalność sterowań symetrycznych (problem Reedsa-Sheppa) oraz niezależność pól wektorowych  $\mathbf{g}_1, \mathbf{g}_2$  od współrzędnych pozycyjnych, możliwe jest znalezienie symetrii rozwiązań. Istnienie symetrii istotnie ułatwia poszukiwanie rozwiązania problemu.

## 10.2 Metoda sterowań sinusoidalnych

Aczkolwiek rozwiązania analityczne zadania planowania ruchu dla ogólnych układów nieholonomicznych nie są znane, jednak metody analityczne okazały się skuteczne dla pewnych podklas układów nieholonomicznych. Aby zrozumieć znaczenie *sterowań sinusoidalnych* układów nieholonomicznych, rozważmy układ (integrator) Brocketta

$$\dot{\mathbf{q}} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ -y \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 1 \\ x \end{pmatrix} u_2 = \mathbf{g}_1(\mathbf{q})u_1 + \mathbf{g}_2(\mathbf{q})u_2. \quad (10.13)$$

Szczególność tego układu wynika z jego specjalnej struktury. Znikanie nawiasów Liego trzeciego stopnia wskazuje, że układ Brocketta jest nilpotentny rzędu 2. Rozważmy zadanie przeprowadzenia układu Brocketta

z punktu  $(0, 0, 0)^T$  do punktu  $(0, 0, z(1))^T$ ,  $z(1) > 0$ , przy minimalizacji energetycznej funkcji kryterialnej  $\mathcal{J}(\mathbf{u}(\cdot)) = \int_0^1 \|\mathbf{u}(t)\|^2 dt$ . Ruch w kierunku osi  $Z$  zasługuje na szczególną uwagę, bowiem biorąc liniową kombinację sterowań stałych  $u_1, u_2$  możemy z punktu  $\mathbf{q}_0 = (0, 0, 0)^T$  osiągnąć każdy punkt płaszczyzny  $XY$  rozpiętej przez wektory  $\mathbf{g}_1(\mathbf{q}_0), \mathbf{g}_2(\mathbf{q}_0)$ . Zatem, sterowania wyprowadzające stan poza tę płaszczyznę muszą być bardziej skomplikowane.

Po raz kolejny wykorzystamy Zasadę Maksimum Pontriagina. Hamiltonian układu przyjmuje postać

$$\begin{aligned} H(\mathbf{q}, \mathbf{p}, \mathbf{u}) &= -\|\mathbf{u}\|^2 + \langle \mathbf{p}, \dot{\mathbf{q}} \rangle = \\ &= -(u_1^2 + u_2^2) + p_1 u_1 + p_2 u_2 + p_3(-y u_1 + x u_2). \end{aligned} \quad (10.14)$$

Zapiszmy równania dołączone

$$\begin{cases} \dot{p}_1 = -\frac{\partial H}{\partial x} = -p_3 u_2 \\ \dot{p}_2 = -\frac{\partial H}{\partial y} = p_3 u_1 \\ \dot{p}_3 = -\frac{\partial H}{\partial z} = 0, \end{cases} \quad (10.15)$$

z których dostajemy w szczególności, że  $p_3 = \text{const}$ . Warunek konieczny wynikający z Zasady Maksimum Pontriagina wiedzie do następującej zależności sterowań od stanu i zmiennych dołączonych

$$\begin{cases} u_1 = \frac{p_1 - p_3 y}{2} \\ u_2 = \frac{p_2 + p_3 x}{2}. \end{cases} \quad (10.16)$$

Korzystając z równań (10.15), (10.16) i równań dołączonych wyliczamy

$$\begin{aligned} \frac{d}{dt} \left( \frac{1}{2} \|\mathbf{u}\|^2 \right) &= u_1 \frac{\dot{p}_1 - p_3 \dot{y}}{2} + u_2 \frac{\dot{p}_2 + p_3 \dot{x}}{2} = \\ &= u_1 \frac{-p_3 u_2 - p_3 u_2}{2} + u_2 \frac{p_3 u_1 + p_3 u_1}{2} = 0, \end{aligned}$$

co oznacza, że energia sterowań jest stała,  $\|\mathbf{u}\|^2 = \text{const}$ , w każdej chwili



czasu. Poszukiwane sterowania wygodnie jest opisać we współrzędnych biegunowych

$$u_1(t) = \dot{x} = r \cos \alpha(t), \quad u_2(t) = \dot{y} = r \sin \alpha(t). \quad (10.17)$$

Uwzględniając stałość  $r$  oraz równania dołączone wyliczamy

$$\ddot{x}^2 + \ddot{y}^2 = r^2 \dot{\alpha}^2 = p_3^2 \|\mathbf{u}\|^2 = \text{const.}$$

Ze względu na stałość  $p_3$ , także  $\dot{\alpha} = \text{const}$ , tzn.

$$\alpha(t) = \lambda t + \alpha_0.$$

Całkując (10.17) i uwzględniając warunki brzegowe  $x(0) = x(1) = 0$ ,  $y(0) = y(1) = 0$  uzyskujemy  $\lambda = 2k\pi$ ,  $k \in \mathbb{Z}$  i  $\alpha_0 = 0$ . Wybierając  $k = 1$  minimalizujemy energię sterowań, a trajektoria będąca rozwiązaniem postawionego zadania jest następująca

$$\begin{cases} x(t) = \frac{r}{2\pi} \sin(2\pi t) \\ y(t) = \frac{r}{2\pi} (1 - \cos(2\pi t)) \\ z(t) = \frac{r^2}{2\pi} \left( t - \frac{1}{2\pi} \sin(2\pi t) \right), \quad t \in [0, 1]. \end{cases} \quad (10.18)$$

Amplitudę sterowań  $u_1$ ,  $u_2$  uzyskuje się z wyrażenia na  $z(t)$  w układzie równań (10.18) dla czasu  $t = 1$

$$r = \sqrt{2\pi z(1)}.$$

Ostatecznie, optymalne sterowania są sinusoidalne

$$\begin{cases} u_1(t) = \sqrt{2\pi z(1)} \cos(2\pi t) \\ u_2(t) = \sqrt{2\pi z(1)} \sin(2\pi t), \quad t \in [0, 1]. \end{cases} \quad (10.19)$$

Rozważony przez nas przykład wskazuje, że optymalnych sterowań dla układu Brocketta (10.13) należy poszukiwać wśród funkcji harmoniczych. Okazuje się, że także dla tzw. *układów łańcuchowych* pierwszego rzędu, opisanych równaniami

$$\begin{cases} \dot{x}_i = u_i \\ \dot{x}_{ij} = x_i u_j, \end{cases} \quad (10.20)$$

$i = 1, \dots, m$ ,  $i > j = 1, \dots, m$ , sterowania sinusoidalne znajdują zastososo-

wanie przy planowaniu ruchu. Dla tych układów, z warunkami brzegowymi  $x_i(0) = x_i(1)$  nałożonymi na część współrzędnych stanu i z unormowanym czasem sterowania równym 1, sterowania optymalne minimalizujące wskaźnik jakości  $\int_0^1 \|\mathbf{u}(t)\|^2 dt$  są sinusoidalne o częstotliwościach  $2k\pi$ ,  $k = 1, \dots, m/2$ . Niech sterowania  $u_i$  będą postaci

$$u_i(t) = \sin(\omega_i t), \quad \omega_i = 2k_i\pi, \quad k_i = 1, \dots, \frac{m}{2}. \quad (10.21)$$

Rozwiązując pierwsze z równań (10.20) dla współrzędnych  $x_i$  mamy

$$x_i(1) = x_i(0) + \int_0^1 \sin(\omega_i t) dt = x_i(0),$$

a zatem współrzędne  $x_i$  pod wpływem sterowań sinusoidalnych  $u_i$  kreślą pętlę. Wyliczając  $x_{ij}$  uzyskujemy

$$\begin{aligned} x_{ij}(1) &= x_{ij}(0) + \int_0^1 -\frac{1}{\omega_i} \cos(\omega_i t) \sin(\omega_j t) dt = \\ &= x_{ij}(0) + \frac{1}{2\omega_i} \int_0^1 (\sin((\omega_i - \omega_j)t) - \sin((\omega_i + \omega_j)t)) dt. \end{aligned} \quad (10.22)$$

Współrzędne  $x_{ij}$  mogą osiągać wartości różne od początkowych, jeśli tylko  $\omega_i \pm \omega_j = 0$ . Jeśli końcowa wartość współrzędnej  $x_{ij}$  jest znana, wtedy łatwo sformułować warunek na częstotliwości  $\omega_i$ ,  $\omega_j$  zapewniający osiągnięcie tej wartości\*.

Planowanie ruchu układów łańcuchowych pierwszego rzędu zadanych równaniami (10.20) staje się teraz oczywiste. Najpierw należy przeprowadzić współrzędne  $x_i$  do ich wartości docelowych, nie bacząc na zmiany współrzędnych  $x_{ij}$ . Następnie, z odpowiednio wyselekcjonowanymi częstotliwościami należy spowodować, by współrzędne  $x_{ij}$  osiągnęły ich finalne wartości bez zmiany współrzędnych  $x_i$ . Procedurę tę można uogólnić na układy łańcuchowe drugiego rzędu opisane równaniami

$$\begin{cases} \dot{x}_i = u_i, & i = 1, \dots, m, \\ \dot{x}_{ij} = x_i u_j, & 1 \leq i < j \leq m, \\ \dot{x}_{ijk} = x_{ij} u_k, & (\text{modulo tożsamość Jacobiego}). \end{cases} \quad (10.23)$$

W powyższym zapisie „modulo tożsamość Jacobiego” oznacza, że jeśli przyjmiemy za dopuszczalne w trzecim równaniu z (10.23) trójki indeksów  $ijk$

\*Być może po zakreśleniu kilku pętli.

oraz  $k_{ij}$ , to nie jest dopuszczalna trójka  $jki$ <sup>†</sup>.

Planowanie ruchu układów łańcuchowych drugiego rzędu, po wykonaniu dwóch kroków jak dla układów łańcuchowych rzędu pierwszego, wymaga jeszcze jednego kroku. Krok ten zapewnia osiągnięcie przez współrzędne  $x_{ijk}$  ich docelowych wartości, bez zmiany wcześniej doprowadzonych do właściwych wartości współrzędnych  $x_i$  i  $x_{ij}$ . Opisaną procedurę ilustruje w sposób poglądowy rysunek 10.3.

Sterowania sinusoidalne wykazały swą użyteczność także przy planowaniu ruchu układów (wielo-)łańcuchowych. Wykazano, że poprzez lokalne sprzężenie zwrotne każdy 2-wejściowy układ nieholonomiczny można przekształcić do postaci 2-łańcuchowej

$$\begin{cases} \dot{x}_0 = u_1 & \dot{y}_0 = u_2 \\ \dot{x}_1 = y_0 u_1 & (\dot{y}_1 = x_0 u_2) \\ \dot{x}_2 = x_1 u_1 & \dot{y}_2 = y_1 u_2 \\ \dot{x}_3 = x_2 u_1 & \dot{y}_3 = y_2 u_2 \\ \vdots & \vdots \\ \dot{x}_{n_x} = x_{n_x-1} u_1 & \dot{y}_{n_y} = y_{n_y-1} u_2. \end{cases} \quad (10.24)$$

Równanie na  $\dot{y}_1$  może być opuszczone, bowiem  $y_1$  jest zmienną zależną ( $y_1 = x_0 y_0 - x_1$ ). Sterowania sinusoidalne znajdują również zastosowanie do planowania ruchu robotów mobilnych z przyczepami.

Podsumujmy teraz zalety i wady metody sterowań sinusoidalnych przy planowaniu ruchu układów nieholonomicznych. Jej zaletą z pewnością jest to, że metoda ma przejrzystą interpretację geometryczną, a jej podstawy tkwią w idei dekompozycji zadań wielowymiarowych na sekwencję zadań w przestrzeniach o mniejszych wymiarach. Ponadto generowane sterowania są ciągłe, co ma znaczenie przy fizycznej realizacji sterowań przez urządzenia techniczne. Za istotną wadę uznać należy fakt, że skuteczność metody jest zapewniona tylko dla bardzo szczególnej klasy systemów; nie każdy układ nieholonomiczny da się sprowadzić do postaci łańcuchowej.

<sup>†</sup>Mnemonotechniczna reguła określania dopuszczalnych trójek indeksów opiera się na zasadzie trójelementowego rejestru cyklicznego. Dopuszczalne są trójki odpowiadające dwóm dowolnym stanom rejestru.

etap	$x_i$	$x_{ij}$	$x_{ijk}$
I	$x_i^0 \rightarrow x_i^f$	$x_{ij}^0 \rightsquigarrow x_{ij}^I$	$x_{ijk}^0 \rightsquigarrow x_{ijk}^I$
II	$x_i^f \circlearrowleft$	$x_{ij}^I \rightarrow x_{ij}^f$	$x_{ijk}^I \rightsquigarrow x_{ijk}^{II}$
III	$x_i^f \circlearrowleft$	$x_{ij}^f \circlearrowleft$	$x_{ijk}^{II} \rightarrow x_{ijk}^f$

Rysunek 10.3 Etapy osiągania zadanych wartości przez współrzędne układu drugiego rzędu.

### 10.3 Metoda bazująca na twierdzeniu Stokesa

Metoda planowania ruchu układów nieholonomicznych bazująca na twierdzeniu Stokesa wykorzystuje możliwość dekompozycji wektora stanu pewnej podklasy układów nieholonomicznych na dwie podgrupy: współrzędnych sterowanych bezpośrednio i współrzędnych sterowanych pośrednio, przez oddziaływanie na pierwszą podgrupę współrzędnych. Osiągnięcie wartości zadanych przez współrzędne sterowane bezpośrednio jest łatwym zadaniem sterowania. Do wymuszenia zadanych wartości współrzędnych sterowanych pośrednio dobieramy sterowania w ten sposób, by współrzędne sterowane bezpośrednio zakreślały parametryzowaną pętlę. Po zakreśleniu pętli, współrzędne sterowane bezpośrednio wracają do początkowych wartości, natomiast zmiana wartości współrzędnych sterowanych pośrednio zależy od parametrów opisujących pętlę. Wymaganą zmianę współrzędnych sterowanych pośrednio uzyskujemy przez rozwiązanie równań algebraicznych, których niewiadomymi są parametry pętli.

Przed przedstawieniem tej metody planowania przypomnimy twierdzenie Stokesa.

**Twierdzenie 10.3.1** *Niech  $S(x, y, z)$  będzie płaszczyzną gładką, zorientowaną powierzchnią o brzegu tworzącym krzywą zamkniętą  $C$  złożoną z odcinkami gładkich krzywych. Niech pole wektorowe  $\mathbf{X} = (X_1, X_2, X_3)^T$  będzie ciągłe wraz z pierwszą pochodną w pewnym obszarze zawierającym  $S$ . Wtedy*

$$\oint_C (X_1 dx + X_2 dy + X_3 dz) = \iint_S \left( \left( \frac{\partial X_3}{\partial y} - \frac{\partial X_2}{\partial z} \right) \cos \alpha + \left( \frac{\partial X_1}{\partial z} - \frac{\partial X_3}{\partial x} \right) \cos \beta + \left( \frac{\partial X_2}{\partial x} - \frac{\partial X_1}{\partial y} \right) \cos \gamma \right) dS,$$

przy czym  $C$  jest trawersowane w kierunku dodatnim (wnętrze obszaru po lewej stronie), a  $\alpha, \beta, \gamma$  są kątami utworzonymi przez normalną do  $S$  z osiami współrzędnych układu  $XYZ$ .

Poniżej wykorzystamy także pewien algebraiczny warunek konieczny całkowalności (holonomiczności) ograniczeń, wynikający z twierdzenia 2.2.1.

**Twierdzenie 10.3.2 Ograniczenie**

$$\mathbf{X}^T \dot{\mathbf{q}} = 0 = \langle (X_1, \dots, X_n)^T, (\dot{q}_1, \dots, \dot{q}_n)^T \rangle \quad (10.25)$$

jest całkowalne (holonomiczne), jeżeli jest spełniony zestaw równań

$$X_\nu \left( \frac{\partial X_\mu}{\partial q_\lambda} - \frac{\partial X_\lambda}{\partial q_\mu} \right) + X_\mu \left( \frac{\partial X_\lambda}{\partial q_\nu} - \frac{\partial X_\nu}{\partial q_\lambda} \right) + X_\lambda \left( \frac{\partial X_\nu}{\partial q_\mu} - \frac{\partial X_\mu}{\partial q_\nu} \right) = 0, \quad (10.26)$$

dla  $\lambda, \mu, \nu = 1, 2, \dots, n$ . Jeżeli warunek (10.26) nie jest spełniony, to ograniczenie (10.25) jest niecałkowalne (nieholonomiczne).

Wyprowadzimy teraz powyższe twierdzenie z definicji ograniczeń holonomicznych. Holonomiczność ograniczenia (10.25) jest równoważna istnieniu czynnika całkującego  $\vartheta(\mathbf{q}) \neq 0$  i pewnej funkcji  $\varphi$  spełniających równanie

$$\begin{aligned} \vartheta(\mathbf{q})(X_1 dq_1 + X_2 dq_2 + \dots + X_n dq_n) &= \\ &= d\varphi = \frac{\partial \varphi}{\partial q_1} dq_1 + \dots + \frac{\partial \varphi}{\partial q_n} dq_n, \end{aligned}$$

czyli

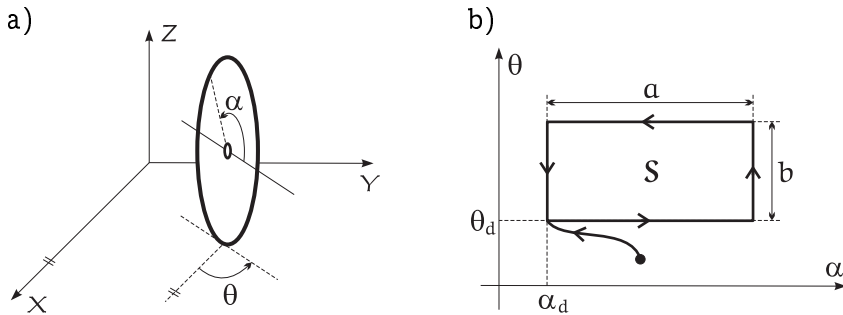
$$\vartheta(\mathbf{q})X_i = \frac{\partial \varphi}{\partial q_i}, \quad i = 1, \dots, n. \quad (10.27)$$

Wybieramy współrzędne  $q_\lambda, q_\mu, q_\nu$  i różniczkujemy warunki (10.27) o numerach  $\lambda, \mu, \nu$ :

$$\begin{cases} \frac{\partial \vartheta}{\partial q_\mu} X_\lambda + \vartheta \frac{\partial X_\lambda}{\partial q_\mu} = \frac{\partial^2 \varphi}{\partial q_\mu \partial q_\lambda} & \frac{\partial \vartheta}{\partial q_\nu} X_\lambda + \vartheta \frac{\partial X_\lambda}{\partial q_\nu} = \frac{\partial^2 \varphi}{\partial q_\nu \partial q_\lambda} \\ \frac{\partial \vartheta}{\partial q_\lambda} X_\mu + \vartheta \frac{\partial X_\mu}{\partial q_\lambda} = \frac{\partial^2 \varphi}{\partial q_\lambda \partial q_\mu} & \frac{\partial \vartheta}{\partial q_\nu} X_\mu + \vartheta \frac{\partial X_\mu}{\partial q_\nu} = \frac{\partial^2 \varphi}{\partial q_\nu \partial q_\mu} \\ \frac{\partial \vartheta}{\partial q_\lambda} X_\nu + \vartheta \frac{\partial X_\nu}{\partial q_\lambda} = \frac{\partial^2 \varphi}{\partial q_\lambda \partial q_\nu} & \frac{\partial \vartheta}{\partial q_\mu} X_\nu + \vartheta \frac{\partial X_\nu}{\partial q_\mu} = \frac{\partial^2 \varphi}{\partial q_\mu \partial q_\nu}. \end{cases}$$

Teraz wykorzystamy równość pochodnych mieszanych funkcji  $\varphi$

$$\begin{cases} \frac{\partial \vartheta}{\partial q_\mu} X_\lambda + \vartheta \frac{\partial X_\lambda}{\partial q_\mu} = \frac{\partial \vartheta}{\partial q_\lambda} X_\mu + \vartheta \frac{\partial X_\mu}{\partial q_\lambda} \\ \frac{\partial \vartheta}{\partial q_\nu} X_\lambda + \vartheta \frac{\partial X_\lambda}{\partial q_\nu} = \frac{\partial \vartheta}{\partial q_\lambda} X_\nu + \vartheta \frac{\partial X_\nu}{\partial q_\lambda} \\ \frac{\partial \vartheta}{\partial q_\nu} X_\mu + \vartheta \frac{\partial X_\mu}{\partial q_\nu} = \frac{\partial \vartheta}{\partial q_\mu} X_\nu + \vartheta \frac{\partial X_\nu}{\partial q_\mu}. \end{cases} \quad (10.28)$$



Rysunek 10.4 a) Koło toczące się po płaskiej powierzchni, b) Idea sterowania współrzędnych sterowanych pośrednio.

Pierwsze równanie w układzie równań (10.28) mnożymy przez  $X_\nu$ , drugie przez  $-X_\mu$ , a trzecie przez  $X_\lambda$ . Wynik sumujemy eliminując pochodne cząstkowe funkcji  $\vartheta$ , a także zależność od funkcji  $\vartheta$ , która z założenia jest różna od zera, co prowadzi do

$$X_\nu \frac{\partial X_\lambda}{\partial q_\mu} - X_\nu \frac{\partial X_\mu}{\partial q_\lambda} - X_\mu \frac{\partial X_\lambda}{\partial q_\nu} + X_\mu \frac{\partial X_\nu}{\partial q_\lambda} + X_\lambda \frac{\partial X_\mu}{\partial q_\nu} - X_\lambda \frac{\partial X_\nu}{\partial q_\mu} = 0. \quad (10.29)$$

Po pomnożeniu obu stron równości (10.29) przez  $-1$  i zgrupowaniu odpowiednich składników otrzymujemy warunek konieczny holonomiczności ograniczeń (10.25).

Zauważmy, że przyjmując  $\vartheta \equiv 1$  uzyskujemy warunek wystarczający holonomiczności ograniczeń (10.26)

$$\frac{\partial X_\mu}{\partial q_\lambda} - \frac{\partial X_\lambda}{\partial q_\mu} = 0, \quad \lambda, \mu = 1, \dots, n. \quad (10.30)$$

Dla przypadku  $n = 3$  łatwo zauważyć, korzystając z twierdzenia Stokesa i definicji ograniczeń holonomicznych, że całkowanie wzdłuż pętli  $\mathcal{C}$  układu o więzach nieholonomicznych daje niezerową wartość całki krzywoliniowej, zatem następuje przemieszczenie stanu po przejściu pętli  $\mathcal{C}$ .

Poniżej przeanalizujemy, jak metoda bazująca na twierdzeniu Stokesa może być zastosowana do zadania planowania ruchu dla koła o promieniu  $r$  toczącego się po płaszczyźnie  $XY$ , pokazanego na rysunku 10.4a.

### Przykład 10.3.1 (Toczenie się koła bez poślizgu)

Toczące się koło o współrzędnych uogólnionych  $(x, y, \theta, \alpha)^T$  podlega ograniczeniom nieholonomicznym wynikającym z braku poślizgu poprzecznego

i wzdłużnego

$$\begin{cases} \dot{x} - r \cos \theta \dot{\alpha} = 0 \\ \dot{y} - r \sin \theta \dot{\alpha} = 0, \end{cases} \quad (10.31)$$

które prowadzą do modelu kinematyki w postaci

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\alpha} \end{pmatrix} = \begin{pmatrix} r \cos \theta \\ r \sin \theta \\ 0 \\ 1 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} u_2, \quad (10.32)$$

ze sterowaniami  $u_1 = \dot{\alpha}$  i  $u_2 = \dot{\theta}$ . Nieholonomiczność pierwszego z ograniczeń (10.31), po wyliczeniu pochodnych

$$\frac{\partial X_1}{\partial \cdot} = \frac{\partial X_2}{\partial \cdot} = \frac{\partial X_3}{\partial \cdot} = \frac{\partial X_4}{\partial x} = \frac{\partial X_4}{\partial y} = \frac{\partial X_4}{\partial \alpha} = 0, \quad \frac{\partial X_4}{\partial \theta} = r \sin \theta,$$

wynika z niespełnienia równania (10.26) dla  $\nu = 1$ ,  $\mu = 4$ ,  $\lambda = 3$ . Analogicznie pokazuje się nieholonomiczność drugiego z równań (10.31).

Algorytm metody bazującej na twierdzeniu Stokesa składa się z trzech kroków.

**Krok 1.** Wprowadzić stan początkowy  $(x_0, y_0, \theta_0)^T$  i końcowy  $(x_d, y_d, \theta_d)^T$ .

**Krok 2.** Sprowadzić zmienne  $\alpha$ ,  $\theta$  sterowane bezpośrednio od wartości początkowych  $\alpha_0$ ,  $\theta_0$  do wartości docelowych  $\alpha_d$ ,  $\theta_d$ . W tym czasie sterowane pośrednio współrzędne  $x$ ,  $y$  osiągną nowe wartości:  $x_0 \rightarrow x_m$ ,  $y_0 \rightarrow y_m$ .

**Krok 3.** Wybrać sparametryzowaną rodzinę krzywych zamkniętych (pętli) w aspekcie współrzędnych  $\alpha\theta$ . Dobrać tak parametry krzywych, by podczas trawersowania pętli uzyskać docelowe wartości zmiennych zależnych. Z definicji realizacji pętli we współrzędnych sterowanych bezpośrednio, ich wartości po zakreśleniu pętli powracają do początkowych. Zadanie planowania ruchu jest rozwiązane.

Rozwiążemy teraz zadanie planowania ruchu dla modelu (10.32). W tym celu, na płaszczyźnie  $\alpha\theta$  wybieramy rodzinę pętli w postaci prostokątów o bokach  $a$ ,  $b$ , z lewym dolnym wierzchołkiem w punkcie  $(\alpha_d, \theta_d)^T$  (zobacz rysunek 10.4b). Równania wynikające z twierdzenia Stokesa dla obsza-

ru  $\mathbf{S} = [\alpha_d, \alpha_d + a] \times [\theta_d, \theta_d + b]$  są następujące

$$\begin{aligned} \oint_C (-r \cos \theta d\alpha + 1 dx) &= \iint_S r \sin \theta d\alpha d\theta = \\ &= \int_{\alpha_d}^{\alpha_d+a} \int_{\theta_d}^{\theta_d+b} r \sin \theta d\alpha d\theta = ar \int_{\theta_d}^{\theta_d+b} \sin \theta d\theta = \\ &= ar (\cos \theta_d - \cos(\theta_d + b)) = 2ar \sin\left(\frac{b}{2}\right) \sin\left(\theta_d + \frac{b}{2}\right). \end{aligned}$$

Zatem, zmiana wartości współrzędnej  $x$  wynosi ( $\alpha$  po zakreśleniu pętli wraca do wartości  $\alpha_d$ )

$$x_d - x_m = 2ar \sin\left(\frac{b}{2}\right) \sin\left(\theta_d + \frac{b}{2}\right). \quad (10.33)$$

Analogicznie, dla drugiego ograniczenia

$$\begin{aligned} \oint_C (-r \sin \theta d\alpha + 1 dy) &= \iint_S -r \cos \theta d\alpha d\theta = \\ &= \int_{\alpha_d}^{\alpha_d+a} \int_{\theta_d}^{\theta_d+b} -r \cos \theta d\alpha d\theta = -ar \int_{\theta_d}^{\theta_d+b} \cos \theta d\theta = \\ &= -ar (\sin(\theta_d + b) - \sin \theta_d) = -2ar \sin\left(\frac{b}{2}\right) \cos\left(\theta_d + \frac{b}{2}\right). \end{aligned}$$

Przyrost sterowanej bezpośrednio współrzędnej  $y$  wynosi

$$y_d - y_m = -2ar \sin\left(\frac{b}{2}\right) \cos\left(\theta_d + \frac{b}{2}\right). \quad (10.34)$$

Z równań (10.33) i (10.34) uzyskujemy warunki na parametry  $a$ ,  $b$

$$\begin{cases} b = 2(-\theta_d + \text{atan2}(y_d - y_m, x_d - x_m)), & 0 < b < 4\pi, \\ a = \frac{\sqrt{(y_d - y_m)^2 + (x_d - x_m)^2}}{2r \sin(b/2)}. \end{cases} \quad (10.35)$$

Zakładając  $\alpha_d > \alpha_0$  i  $\theta_d > \theta_0$ , przykładowe sterowania rozwiązujące zadanie



planowania ruchu koła są postaci

$$\begin{cases} u_1 = 1 & u_2 = 0, & t = [0, \alpha_d - \alpha_0] = [0, t_1], \\ u_1 = 0 & u_2 = 1, & t = [t_1, t_1 + \theta_d - \theta_0] = [t_1, t_2], \\ u_1 = 1 & u_2 = 0, & t = [t_2, t_2 + a] = [t_2, t_3], \\ u_1 = 0 & u_2 = 1, & t = [t_3, t_3 + b] = [t_3, t_4], \\ u_1 = -1 & u_2 = 0, & t = [t_4, t_4 + a] = [t_4, t_5], \\ u_1 = 0 & u_2 = -1, & t = [t_5, t_5 + b] = [t_5, T], \end{cases}$$

gdzie parametry  $a$  i  $b$  wyliczamy ze wzoru (10.35), w którym za  $x_m$ ,  $y_m$  należy podstawić

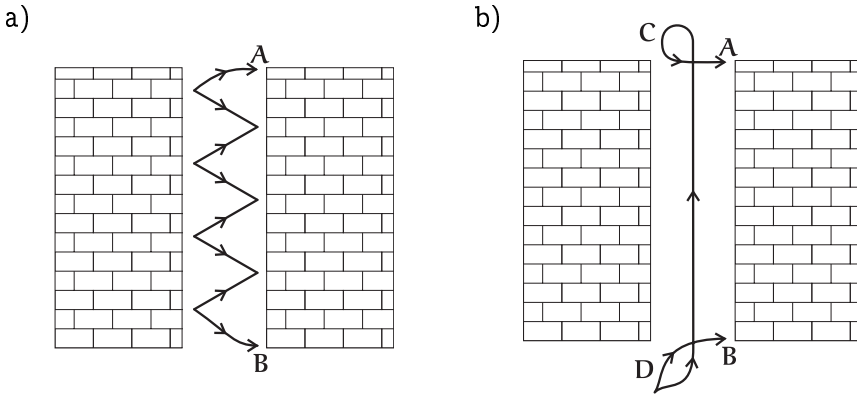
$$\begin{cases} x_m = x_0 + r(\sin \theta_d - \sin \theta_0) \\ y_m = y_0 + r(\cos \theta_0 - \sin \theta_d). \end{cases}$$

■

Wadą metody bazującej na twierdzeniu Stokesa jest ograniczenie wymiaru rozwiązywanych zadań planowania ruchu.

## 10.4 Metoda osiągnięcia podcelów

Oprócz przedstawionych dotąd metod planowania ruchu można zaproponować inne metody, bazujące na wykorzystaniu struktury modelu układu nieholonomicznego. Rozważmy jednokołowy robot mobilny opisany równaniami (8.8). Trójwymiarowa przestrzeń stanu robota daje się zdekomponować na dwuwymiarową podprzestrzeń położeń  $(x, y)^T$  i jednowymiarową podprzestrzeń orientacji  $\theta$ . Sterowanie  $u_2$  działając jedynie na współrzędną orientacji nie zmienia współrzędnych położeniowych, czyli zadanie planowania ruchu jest praktycznie rozwiązane, gdy robot osiągnie zadaną pozycję. Zatem zadanie planowania ruchu jest dekomponowalne na dwa podzadania (podcele): osiągnięcia najpierw zadanej pozycji  $(x, y)^T$ , a następnie zadanej orientacji  $\theta$ . Podcel osiągnięcia zadanej pozycji najłatwiej zrealizować przez ruch prostoliniowy opisany sterowaniami  $u_1 = \text{const} \neq 0$ ,  $u_2 = 0$ . Z kolei, aby ruch prostoliniowy przeprowadził położenie początkowe w docelowe, należy osiągnąć podcel polegający na odpowiedniej zmianie orientacji początkowej robota,  $u_1 = 0$ ,  $u_2 = \text{const} \neq 0$ . Metoda osiągnięcia podcelów nie daje się zaliczyć ani do metod globalnych, ani do metod lokalnych. Można ją raczej zaliczyć do metod sztucznej inteligencji niż „twardych” metod analityczno- optymalizacyjnych. Geometrycznie metodę podcelów



**Rysunek 10.5** a) Metoda lokalna zapewniająca monotoniczne ubywanie odległości od stanu docelowego. b) Metoda osiągania podcelów. Głównym podcelem jest ruch równoległy do brzegu przeszkód. Podczas realizacji ruchu odległość od stanu docelowego może się zwiększać, np. w punktach C i D.

można opisać jako konstrukcję zbioru przecinających się podprzestrzeni. Ruch w obrębie każdej z nich powinien być łatwy do realizacji przy pomocy sterowań. Na przecięciu podprzestrzeni leżą podcele, których osiągnięcie przybliży realizację celu, choć przybliżanie to niekoniecznie oznacza zmniejszenie odległości do celu. Wadą metody osiągania podcelów w planowaniu ruchu jest wąski zakres stosowalności i konieczność dokładnej analizy struktury modelu układu. Może się okazać, że nie uda się zdekomponować zadania planowania ruchu na podzadania i wtedy należy stosować inne metody planowania ruchu. Wydaje się także, że definiowanie podcelów nie daje się zautomatyzować, a wykorzystywane przy nim spostrzegawczość i doświadczenie człowieka są trudne do zastąpienia.

Wada metod lokalnych wymagających lokalnego zmniejszania odległości euklidesowej do celu jest widoczna na rysunku 10.5. Realizacja trajektorii łączącej punkty A i B zgodnie z zaleceniami metod lokalnych jest przedstawiona na rysunku 10.5a. Jeśli dla przeorientowania robota wykorzystamy wolną przestrzeń i dopuścimy chwilowe zwiększenie odległości od stanu docelowego, to sumaryczna długość trajektorii zobrazowanej na rysunku 10.5b jest mniejsza od długości trajektorii z rysunku 10.5a.

W niniejszym podrozdziale zaprezentujemy metodę generacji podcelów dla dwóch typów robotów mobilnych: jednokołowego robota mobilnego i samochodu kinematycznego.

Założmy, że amplitudy sterowań są ograniczone  $|u_1|, |u_2| \leq 1$ . Do pre-

zencacji algorytmów planowania ruchu wymagane są definicje odległości kątowej i normalizacji kąta. Odległość kątowa między kątami  $\alpha, \beta \in [0, 2\pi)$  wynosi

$$\|\alpha - \beta\| = \min(2\pi - |\alpha - \beta|, |\alpha - \beta|).$$

Normalizacja kąta  $\gamma$  sprowadza jego wartość do zakresu  $[0, 2\pi)$ , tj.  $\gamma_{\text{norm}} = \gamma \bmod 2\pi$ .

#### Przykład 10.4.1 (Jednokołowy robot mobilny)

Algorytm planowania ruchu dla jednokołowego robota mobilnego składa się z następujących kroków.

**Krok 1.** Wprowadzić stan początkowy  $(x_0, y_0, \theta_0)^T$  i końcowy  $(x_d, y_d, \theta_d)^T$ .

**Krok 2.** Wyliczyć kąt  $\xi$  z zależności  $\xi = \text{atan2}(x_d - x_0, y_d - y_0)$ .

**Krok 3.** Użyć sterowania  $u_2$  do osiągnięcia kąta  $\xi$ , gdy jest spełniona nierówność  $\|\theta_0 - \xi\| < \|\theta_0 - (\xi + \pi) \bmod 2\pi\|$  lub do osiągnięcia kąta  $(\xi + \pi) \bmod 2\pi$  w przeciwnym przypadku. Odpowiednie sterowania są zebrane w tabeli 10.4, która jest indeksowana numerem segmentu, do którego należy kąt  $\xi$ , zobacz rysunek 10.6. Podczas tego ruchu współrzędne  $x, y$  nie zmieniają się.

**Krok 4.** Użyć sterowania  $u_1^*$  przez czas  $\sqrt{(x_d - x_0)^2 + (y_d - y_0)^2}$ , po którym współrzędne  $x, y$  osiągną wartości docelowe  $x_d, y_d$ . W tym kroku orientacja  $\theta$  pozostaje niezmienna.

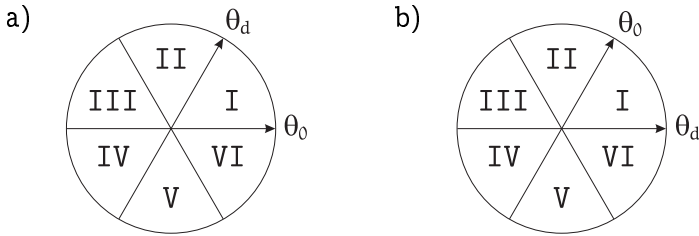
**Krok 5.** Użyć sterowania  $u_2^*$  dopóki  $\theta$  nie osiągnie wartości  $\theta_d$ . Ponieważ współrzędne  $x, y$  nie zmieniają przy takim sterowaniu swych wartości, zadanie planowania ruchu zostaje w ten sposób zrealizowane.

Przedstawiony algorytm planowania ruchu jednokołowego robota mobilnego uwzględnia minimalizację energii zużytej na sterowanie.

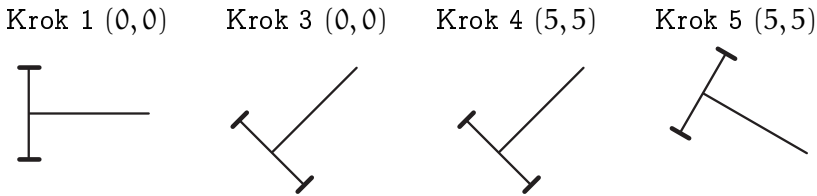
Na rysunku 10.7 pokazano ciąg konfiguracji jednokołowego robota mobilnego, przemieszczającego się zgodnie z algorytmem planowania od stanu  $(x, y, \theta)^T = (0, 0, 0)^T$  do stanu  $(5, 5, -\pi/6)^T$ , w momencie osiągnięcia kolejnych podcelów. ■

---

\*Zobacz tabela 10.4.



**Rysunek 10.6** Numeracja segmentów, do których należy kąt  $\xi$ : a)  $\theta_d < \theta_0$ ; b)  $\theta_0 < \theta_d$ . Szerokość segmentów II, III, V, VI wynosi  $1/2(\pi - \|\theta_0 - \theta_d\|)$ . W zależności od numeru segmentu wyznaczane są sterowania z tabeli 10.4.



**Rysunek 10.7** Położenia i orientacje charakteryzujące podcele dla jednokołowego robota mobilnego podczas rozwiązywania zadania planowania ruchu.

#### Przykład 10.4.2 (Samochód kinematyczny)

Podobny sposób postępowania można zastosować do samochodu kinematycznego opisanego równaniami

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\varphi} \end{pmatrix} = \begin{pmatrix} \cos \varphi \cos \theta \\ \cos \varphi \sin \theta \\ \sin \varphi \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_2,$$

gdzie  $u_1$  jest prędkością liniową kół tylnych, a  $u_2$  opisuje prędkość kątową kół przednich.

Podobnie jak dla jednokołowego robota mobilnego, proces planowania ruchu jest tutaj sparametryzowany zmiennymi decyzyjnymi  $\text{sgn}_i \in \{-1, 1\}$ ,  $i = 1, \dots, 7$  włączającymi sterowania. Algorytm planowania ruchu dla samochodu kinematycznego opisuje następująca sekwencja kroków.

**Krok 1.** Wprowadzić stan początkowy samochodu  $(x_0, y_0, \theta_0, \varphi_0)^T$  i jego stan końcowy  $(x_d, y_d, \theta_d, \varphi_d)^T$ .

**Krok 2.** Wyznaczyć kąt  $\xi$  z zależności  $\xi = \text{atan2}(x_d - x_0, y_d - y_0)$ .

segment	$\theta_0 < \theta_d$				
	Krok				odległość kątowna
	3		4	5	Krok 3 i 5
	$u_2$	do kąta	$u_1$	$u_2$	$\ \theta_0 - \theta_d\ $
I	+1	$\xi$	+1	+1	0
II	+1	$\xi$	+1	-1	$2\ \xi - \theta_d\ $
III	-1	$\xi + \pi$	-1	+1	$2\ \xi + \pi - \theta_0\ $
IV	+1	$\xi + \pi$	-1	+1	0
V	+1	$\xi + \pi$	-1	-1	$2\ \xi + \pi - \theta_d\ $
VI	-1	$\xi$	+1	+1	$2\ \xi - \theta_0\ $

	$\theta_0 > \theta_d$				
	Krok				odległość kątowna
	3		4	5	Krok 3 i 5
	$u_2$	do kąta	$u_1$	$u_2$	$\ \theta_0 - \theta_d\ $
I	-1	$\xi$	+1	-1	0
II	+1	$\xi$	+1	-1	$2\ \xi - \theta_0\ $
III	-1	$\xi + \pi$	-1	+1	$2\ \xi + \pi - \theta_d\ $
IV	-1	$\xi + \pi$	-1	-1	0
V	+1	$\xi + \pi$	-1	-1	$2\ \xi + \pi - \theta_0\ $
VI	-1	$\xi$	+1	+1	$2\ \xi - \theta_d\ $

Tabela 10.4 Wartości sterowań.

- Krok 3.** Zastosować sterowanie  $u_2 = \text{sgn}_1$ : do osiągnięcia przez kąt  $\varphi$  wartości  $\pi/2$ , jeżeli  $\min(\|\varphi_0 - \pi/2\|, \|\varphi_0 + \pi/2\|) = \|\varphi_0 - \pi/2\|$  lub wartości  $-\pi/2$  w przeciwnym przypadku. W tym kroku wszystkie współrzędne, za wyjątkiem  $\varphi$ , zachowują swoje wartości.
- Krok 4.** Użyć sterowania  $u_1 = \text{sgn}_2$  do osiągnięcia przez kąt  $\theta$  wartości  $\xi$ , gdy  $\|\theta_0 - \xi\| < \|\theta_0 - (\xi + \pi) \bmod 2\pi\|$ , w przeciwnym przypadku do osiągnięcia wartości  $(\xi + \pi) \bmod 2\pi$ . Tutaj, oprócz kąta  $\theta$ , zmienia swą wartość tylko kąt  $\varphi$ .
- Krok 5.** Zastosować sterowanie  $u_2 = \text{sgn}_3$ , aż kąt  $\varphi$  osiągnie wartość 0 lub  $\pi$ . Przy realizacji tego kroku pozostałe współrzędne pozostają bez zmian.
- Krok 6.** Zastosować sterowanie  $u_1 = \text{sgn}_4$ , aż pozycja robota  $(x, y)^T$  osiągnie swą wartość docelową  $(x_d, y_d)^T$ .
- Krok 7.** Użyć sterowania  $u_2 = \text{sgn}_5$ , by osiągnąć  $\varphi = \pi/2$ , gdy jest speł-

niona nierówność  $\|\varphi_d - \pi/2\| \leq \|\varphi_d + \pi/2\|$  lub  $\varphi = -\pi/2$ , jeżeli  $\|\varphi_d - \pi/2\| > \|\varphi_d + \pi/2\|$ .

**Krok 8.** Zastosować sterowanie  $u_1 = \text{sgn}_6$ , aż kąt  $\theta$  osiągnie swą wartość docelową  $\theta_d$ . Przy takim sterowaniu pozostałe współrzędne zachowują swe wartości.

**Krok 9.** Użyć sterowania  $u_2 = \text{sgn}_7$ , by kąt  $\varphi$  osiągnął wartość docelową  $\varphi_d$ . Pozostałe współrzędne już osiągnęły swe zadane wartości w poprzednich krokach, a w tym się nie zmieniają, zatem zadanie planowania ruchu zostało rozwiązane.

W każdym kroku algorytmu aktywne jest tylko jedno sterowanie o amplitudzie 1. Zatem optymalizacja energii ruchu

$$\int_0^T (u_1^2(t) + u_2^2(t)) dt = \sum_{i=1}^{seg} \int_0^{T_{seg_i}} 1 dt = \sum_{i=1}^{seg} T_{seg_i} = T,$$

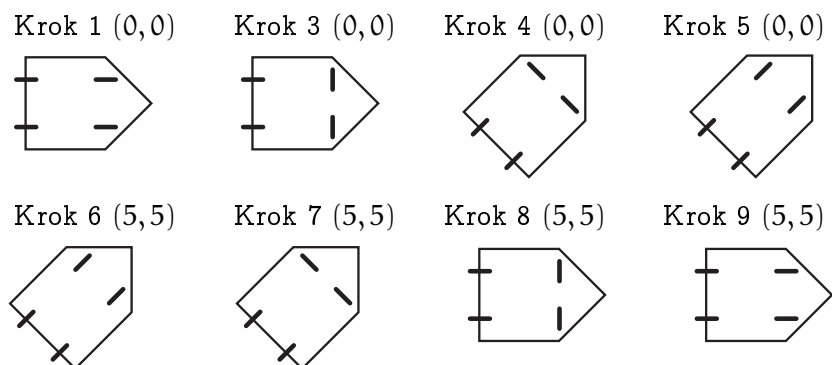
jest równoważna minimalizacji całkowitego czasu ruchu  $T$ . Przedziały  $T_{seg}$  są wyznaczone przez stany układu i zmienne decyzyjne  $\text{sgn}_i$ ,  $i = 1, \dots, 7$ . Energetyczny problem optymalizacji zadany jest równaniem

$$\text{koszt}(\mathbf{sgn}^*) = \min_{\mathbf{sgn} \in \{-1,1\}^7} \text{koszt}(\mathbf{sgn}),$$

gdzie  $\mathbf{sgn} = (\text{sgn}_1, \dots, \text{sgn}_7)$ . Jeśli rozwiązaliśmy zadanie planowania ruchu dla jednokołowego robota mobilnego, to możemy je wykorzystać do wyznaczenia optymalnych sterowań dla samochodu kinematycznego. Zauważamy bowiem, że kroki 4, 6 i 8 algorytmu planowania dla samochodu kinematycznego odpowiadają krokom 3, 4, 5 algorytmu planowania ruchu jednokołowego robota mobilnego. Energia zużywana w krokach 3, 5, 7 i 9 służy zmianie współrzędnej  $\varphi$  i jest równa, odpowiednio,  $\min(\|\varphi_0 - \pi/2\|, \|\varphi_0 + \pi/2\|)$ ,  $\pi/2$ ,  $\pi/2$ ,  $\min(\|\varphi_d - \pi/2\|, \|\varphi_d + \pi/2\|)$ .

Efekt zastosowania przedstawionego algorytmu do planowania ruchu samochodu kinematycznego od stanu początkowego  $(x, y, \theta, \varphi)^T = (0, 0, 0, 0)^T$  do stanu  $(5, 5, 0, 0)^T$  ilustruje rysunek 10.8. ■

Dane do ilustracji omówionych przykładów planowania ruchu zostały wybrane w taki sposób, by odległość euklidesowa do punktów docelowych nie malała monotonicznie. Podczas pierwszej fazy ruchu robotów, odległość ta nawet rośnie. Widzimy zatem, że metoda osiągania podcelów w planowaniu ruchu nie podlega zasadom metod lokalnych, zapewniających zmniejszanie odległości w odpowiednio krótkich przedziałach czasu. Metoda ta nie



**Rysunek 10.8** Położenia i orientacje samochodu kinematycznego po wykonaniu głównych kroków algorytmu planowania ruchu.

jest także klasycznie rozumianą metodą globalną (jak np. Zasada Maksimum czy technika uśredniania). Nie definiuje ona bowiem żadnego wskaźnika jakości zależnego od sterowań, którego minimalizacja zapewni dążenie stanu początkowego do stanu docelowego.

## 10.5 Komentarze i uwagi bibliograficzne

Zadanie planowania ruchu jednokołowego robota mobilnego w środowisku bezkolizyjnym z dopuszczalnym ruchem jedynie do przodu było jednym z pierwszych zadań planowania rozważanym przez Dubinsa w pracy [Dub57]. Pierwsze rozwiązanie problemu Reedsa-Sheppa zawiera praca [RS90], w której liczbę rodzin trajektorii potencjalnie optymalnych ustalono na 48. Późniejsze prace Sussmanna, [ST91] oraz Soueresa, Boissonata i Laumonda, [SL96, SB98], zredukowały tę liczbę do 46. Analiza zadania Reedsa-Sheppa przedstawiona w podrozdziale 10.1 jest wzorowana na pracy [SB98]. Wykorzystanie optymalnych trajektorii, uzyskanych dla jednokołowego robota mobilnego w środowisku bezkolizyjnym, do planowania trajektorii w środowisku z przeszkodami zawiera praca Laumonda i jego współpracowników, [LJTM94]. Jest ona interesującą egzemplifikacją strategii „dziel i rządź” zastosowaną do zadania planowania ruchu. W metodzie Laumonda najpierw planowana jest trajektoria między punktem początkowym a końcowym bez uwzględniania przeszkód. Jeśli wynikowa trajektoria jest kolizyjna, wprowadza się bezkolizyjny punkt pośredni i zadanie planowania jest rozwiązywane dla dwóch segmentów: między punktem początkowym a dodanym punktem

pośrednim i między punktem pośrednim a końcowym. Rekurencyjna procedura podziału zadań planowania musi się zakończyć powodzeniem, jeśli tylko istnieje trajektoria bezkolizyjna, wyznaczona przez układ planowania trajektorii holonomicznych, otoczona trajektoriami bezkolizyjnymi. Na znaczenie sterowań sinusoidalnych w planowaniu ruchu układów nieholonomicznych zwrócił uwagę Brockett [Bro81]. Metodę planowania ruchu przy pomocy sterowań sinusoidalnych upowszechnili dla robotów mobilnych Murray, Sastry, Tilbury i Walsh. Postać łańcuchową modelu kinematyki układu nieholonomicznego można znaleźć w pracach Murraya i współpracowników, [MS93, TLM<sup>+</sup>92, TMS93]. Metoda planowania ruchu bazująca na twierdzeniu Stokesa pochodzi z pracy Mukherjee i Anderson, [MA93]. Metodę osiągania podcelów zaproponowano w pracy [Dul98]. Specjalizowaną metodą planowania ruchu jest metoda Sussmanna-Lafferiera stosowalna do układów nilpotentnych lub nilpotentyzowalnych, [LS91, Laf91]. Metoda ta wykorzystuje istnienie skończonej liczby niezerowych elementów bazy Halla. Choć istnieje technika przybliżonej nilpotentyzacji dowolnego układu nieholonomicznego, [BLC93], to jednak jest to procedura złożona obliczeniowo i lokalna, [SD98]. Metoda Sussmanna-Lafferiera zakłada, że dowolny układ planowania trajektorii holonomicznej wyznacza gładką trajektorię bezkolizyjną, która jest zapisywana we współrzędnych wyznaczonych przez skończony podzbiór bazy Halla. Następnie przy pomocy formuły CBHD jest generowana trajektoria nieholonomiczna o uprzednio wyznaczonych współrzędnych. Zaletą i wadą zarazem metod specjalizowanych jest ograniczenie ich zakresu zastosowania do ściśle określonej podklasy układów nieholonomicznych. Jest to zaleta, bo metody specjalizowane rozwiązują zadanie planowania ruchu w sposób obliczeniowo bardziej efektywny niż metody ogólnego przeznaczenia. Jest to wada, gdyż metody specjalizowane nie uogólniają się na szersze klasy układów nieholonomicznych. W praktyce przy rozwiązywaniu zadań planowania ruchu warto najpierw sprawdzić, czy model, dla którego projektujemy algorytm planowania ruchu nie leży w zakresie stosowalności metod specjalizowanych (lub nie istnieje taka transformacja, która uczyni model podlegającym metodom specjalizowanym). Dopiero gdy taka próba zawiedzie, należy odwoływać się do metod planowania ruchu ogólnego przeznaczenia.

Naszym zdaniem, oprócz sterowań wielomianowych i sinusoidalnych należy poszukiwać innych klas funkcji przydatnych przy sterowaniu układów nieholonomicznych. Na przykład, prace Brocketta idą w kierunku zastosowania funkcji eliptycznych. Wydaje się, że racjonalne postępowanie w za-



kresie metod planowania robotów nieholonomicznych powinno polegać na wyselekcjonowaniu układów, lub klas układów szczególnie ważnych z praktycznego punktu widzenia, a następnie opracowaniu wyspecjalizowanych metod planowania ruchu tych układów.

## Literatura

- [BLC93] A. Bellaïche, J. P. Laumond i M. Chyba, Canonical nilpotent approximation of control systems: Application to nonholonomic motion planning. W: *Proc. IEEE Conf. Decision Control*, vol. 3, strony 2694–2699, San Antonio, 1993.
- [Bro81] R. W. Brockett, Control theory and singular Riemannian geometry. W: P. Hinton i G. Young, (red.), *New Directions in Applied Mathematics*, strony 11–27. Springer-Verlag, New York, 1981.
- [Dub57] L. E. Dubins, On curves of minimal length with a constrained on average curvature and with prescribed initial and terminal positions and tangents. *American J. Math.*, strony 497–516, 1957.
- [Dul98] I. Dułęba, Invariant subspace approach to nonholonomic motion planning. W: *Proc. MMAR Symposium*, vol. 3, strony 921–924, Międzyzdroje, 1998.
- [Laf91] G. Lafferriere, A general strategy for computing steering controls of systems without drift. W: *Proc. IEEE Conf. Decision Control*, vol. 2, strony 1115–1120, Brighton, 1991.
- [LJTM94] J. P. Laumond, P. E. Jacobs, M. Taix i R. M. Murray, A motion planner for nonholonomic mobile robots. *IEEE Trans. Robotics Automat.*, 10(5):577–593, 1994.
- [LS91] G. Lafferriere i H. J. Sussmann, Motion planning for controllable systems without drift. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 2, strony 1148–1153, Sacramento, 1991.
- [MA93] R. Mukherjee i D. P. Andersson, Nonholonomic motion planning using Stokes's theorem. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 802–809, Atlanta, 1993.
- [MS93] R. M. Murray i S. Sastry, Nonholonomic motion planning: Steering using sinusoids. *IEEE Trans. Autom. Contr.*, 38(5):700–716, 1993.
- [RS90] J. A. Reeds i R. A. Shepp, Optimal paths for a car that goes both forward and backwards. *Pacific Journ. Math.*, 145(2):367–393, 1990.

- [SB98] P. Souères i J. D. Boissonnat, Optimal trajectories for nonholonomic mobile robots. W: J. P. Laumond, (red.), *Robot Motion Planning and Control*, vol. 229 serii *Lecture Notes in Control and Information Sciences*. Springer-Verlag, New York, 1998.
- [SD98] J. Sówka i I. Dulęba, Sfery dla nieholonomicznych systemów bezdryfowych. W: *Materiały 5 Krajowej Konferencji Robotyki*, vol. 1, strony 161–169, Świeradów Zdrój, 1998.
- [SL96] P. Souères i J. P. Laumond, Shortest paths synthesis for a car-like robot. *IEEE Trans. Autom. Contr.*, 41(5):672–688, 1996.
- [ST91] H. J. Sussmann i W. Tang, Shortest paths for the Reeds-Sheep car: A worked out example of the use of geometric techniques in nonlinear optimal control. SYCON 10, Rutgers University, 1991. raport wewnętrzny.
- [TLM<sup>+</sup>92] D. Tilbury, J. P. Laumond, R. Murray, S. Sastry i G. Walsh, Steering car-like systems with trailers using sinusoids. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 2, strony 1993–1998, Nice, 1992.
- [TMS93] D. Tilbury, R. Murray i S. Sastry, Trajectory generation for the n-trailer problem using Goursat normal form. W: *Proc. IEEE Conf. Decision Control*, vol. 1, strony 971–977, San Antonio, 1993.

## Rozdział 11

# Modele i algorytmy sterowania kołowych robotów mobilnych

Roboty mobilne poruszające się na kołach nazywamy *kołowymi robotami mobilnymi*. Przyjmujemy, że kołowy robot mobilny składa się z *wózka napędzanego (traktora)* ciągnącego nienapędzane *przyczepy*. Traktor jest wyposażony w silniki, które wprowadzają w ruch cały układ. Robota posiadającego przyczepy nazywamy *złożonym* — kołowy robot mobilny bez przyczep będzie nazywany *prostym*. Zakładamy, że ruch kołowego robota mobilnego odbywa się bez poślizgu kół. Wymaganie to definiuje nieholonomiczne ograniczenia fazowe ruchu układu. Intensywne prace badawcze prowadzone w ostatnich kilku latach zaowocowały powstaniem nowych i specyficznych algorytmów sterowania kołowych robotów mobilnych.

W tym rozdziale książki zajmiemy się metodami modelowania i algorytmami sterowania kołowych robotów mobilnych. Rozpocznijemy od przedstawienia modeli kinematyki i dynamiki prostych i złożonych kołowych robotów mobilnych. Następnie omówimy wybrane algorytmy sterowania. Pierwszym z nich jest algorytm Corona-Pometa, wykorzystujący statyczne sprzężenie zwrotne zależne od czasu. Po nim omówimy algorytm linearyzacji dynamicznej. Kolejnym algorytmem będzie algorytm opracowany przez Walsha, Tilbury’ego, Sastry’ego, Murraya i Laumonda, oparty na przybliżeniu liniowym dynamiki robota w otoczeniu trajektorii zadanej. W dwóch następnych podrozdziałach zajmiemy się ideą sterowania we współrzędnych linearyzujących, realizowaną zarówno w postaci algorytmów nieadaptacyj-

nych, jak i adaptacyjnych. Ostatnim z omawianych algorytmów będzie uniwersalny adaptacyjny  $\lambda$ -śledzący algorytm sterowania.

## 11.1 Dynamika układu robotycznego z ograniczeniami

W podrozdziale 2.2 ustaliliśmy, że kinematyka układu robotycznego określonego na uniwersum fazowym  $\mathbb{R}^{2N}$ , podlegającego  $l$  niezależnym ograniczeniom fazowym typu Pfaffa

$$\mathbf{A}(\mathbf{q})\dot{\mathbf{q}} = \mathbf{0}, \quad (11.1)$$

jest zdefiniowana za pośrednictwem bezdryfowego układu sterowania

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\boldsymbol{\eta}. \quad (11.2)$$

Przy założeniu, że układ robotyczny nie podlega ograniczeniom konfiguracyjnym (liczba ograniczeń konfiguracyjnych  $k = 0$ ), wymiar przestrzeni stanu układu (11.2) wynosi  $n = N$ . Macierz  $\mathbf{G}(\mathbf{q})$ , której kolumny rozpinają przestrzeń zerową macierzy  $\mathbf{A}(\mathbf{q})$  jest rozmiaru  $n \times m$ , gdzie  $m = n - l$ . Jak pokazaliśmy w podrozdziale 2.2, wektor  $\boldsymbol{\eta} \in \mathbb{R}^m$  reprezentuje część składowych wektora prędkości  $\dot{\mathbf{q}}$ . Aby połączyć model dynamiki układu robotycznego (5.2) z zależnością (11.2), zróżniczkujemy (11.2) względem czasu

$$\ddot{\mathbf{q}} = \dot{\mathbf{G}}(\mathbf{q})\boldsymbol{\eta} + \mathbf{G}(\mathbf{q})\dot{\boldsymbol{\eta}} \quad (11.3)$$

i skorzystamy z *Zasady d'Alemberta*, w myśl której siły uogólnione  $\mathbf{F}$  zapewniające spełnienie ograniczeń fazowych (11.1) nie wykonują pracy na dopuszczalnych przemieszczeniach. Z *Zasady d'Alemberta* wynika równość

$$\mathbf{F}^T \dot{\mathbf{q}} = 0,$$

która wobec założonej niezależności ograniczeń fazowych prowadzi do wniosku, że istnieje wektor mnożników Lagrange'a  $\boldsymbol{\lambda} \in \mathbb{R}^l$ , taki że

$$\mathbf{F}^T = \boldsymbol{\lambda}^T \mathbf{A}(\mathbf{q}). \quad (11.4)$$

Po wzięciu pod uwagę zależności (11.4) i (5.3) oraz przy założeniu, że oddziaływania sterujące wpływają na  $m$  stopni swobody układu za pośrednic-

tłem pewnej *macierzy sterowań*  $\mathbf{B}(\mathbf{q})$ , równania dynamiki układu robotycznego z ograniczeniami można zapisać w postaci

$$\begin{aligned} \frac{\partial^2 L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}^2} \ddot{\mathbf{q}} + \frac{\partial^2 L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q} \partial \dot{\mathbf{q}}} \dot{\mathbf{q}} - \frac{\partial L(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}} = \\ = \mathbf{Q}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) = \mathbf{A}^T(\mathbf{q}) \boldsymbol{\lambda} + \mathbf{B}(\mathbf{q}) \mathbf{u}. \end{aligned} \quad (11.5)$$

Macierz sterowań  $\mathbf{B}(\mathbf{q})$  występująca w powyższym wzorze ma rozmiar  $n \times m$  i składa się z elementów  $b_{ij}(\mathbf{q})$ , takich że

$$b_{ij}(\mathbf{q}) = \begin{cases} 1, & \text{jeżeli sterowanie } u_j \text{ działa} \\ & \text{bezpośrednio na współrzędną } q_i, \\ 0, & \text{w przeciwnym przypadku.} \end{cases}$$

W celu wyeliminowania mnożników  $\boldsymbol{\lambda}$  skorzystamy z własności

$$\mathbf{A}(\mathbf{q})\mathbf{G}(\mathbf{q}) = \mathbf{0} \iff \mathbf{G}^T(\mathbf{q})\mathbf{A}^T(\mathbf{q}) = \mathbf{0}.$$

Konsekwencją ostatniej własności oraz równości (11.2) i (11.3) jest następujące sformułowanie równań dynamiki (11.5)

$$\begin{cases} \dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\boldsymbol{\eta} \\ \mathbf{G}^T(\mathbf{q})\mathbf{Q}(\mathbf{q})\mathbf{G}(\mathbf{q})\boldsymbol{\eta} + \mathbf{G}^T(\mathbf{q}) \left( \mathbf{C}(\mathbf{q}, \mathbf{G}(\mathbf{q})\boldsymbol{\eta})\mathbf{G}(\mathbf{q}) + \mathbf{Q}(\mathbf{q})\dot{\mathbf{G}}(\mathbf{q}) \right) \boldsymbol{\eta} + \\ + \mathbf{G}^T(\mathbf{q})\mathbf{D}(\mathbf{q}) = \mathbf{G}^T(\mathbf{q})\mathbf{B}(\mathbf{q})\mathbf{u}. \end{cases} \quad (11.6)$$

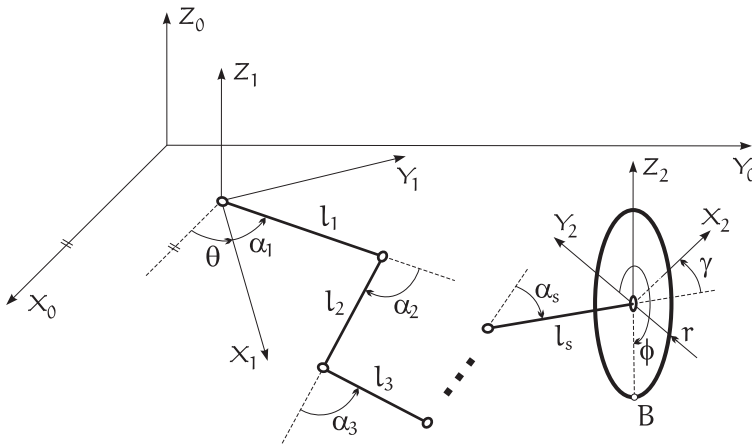
Dysponując trajektorią  $\mathbf{q}(t)$  układu robotycznego, obliczoną jako rozwiązanie układu równań (11.6) przy określonym sterowaniu  $\mathbf{u}(t)$  i warunkach początkowych  $\mathbf{q}(0)$ ,  $\dot{\mathbf{q}}(0)$ , jesteśmy w stanie wyznaczyć wektor mnożników Lagrange'a  $\boldsymbol{\lambda}$ , a co za tym idzie, siły uogólnione  $\mathbf{F}$  zapewniające spełnienie ograniczeń fazowych

$$\begin{cases} \boldsymbol{\lambda} = (\mathbf{A}(\mathbf{q})\mathbf{A}^T(\mathbf{q}))^{-1} \mathbf{A}(\mathbf{q}) (\mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) - \mathbf{B}(\mathbf{q})\mathbf{u}) \\ \mathbf{F} = \mathbf{A}^T(\mathbf{q})\boldsymbol{\lambda}. \end{cases} \quad (11.7)$$

## 11.2 Modele kołowych robotów mobilnych

Założmy, że ruch kołowego robota mobilnego podlega ograniczeniom fazowym (11.1). Podstawowe zadanie modelowania robota mobilnego polega na wyznaczeniu modelu kinematyki (2.36), w formie układu sterowania

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{q})u_i, \quad (11.8)$$



Rysunek 11.1 Schemat kinematyczny pojedynczego koła w układzie  $X_0Y_0Z_0$ .

oraz modelu dynamiki (11.6). W równaniu (11.8) macierz  $\mathbf{G}(\mathbf{q})$  spełnia warunek  $\mathbf{A}(\mathbf{q})\mathbf{G}(\mathbf{q}) = \mathbf{0}$ , natomiast  $\mathbf{u} \in \mathbb{R}^m$  oznacza wektor sterowań\*.

Celem tego podrozdziału będzie wyznaczenie specyficznej postaci modelu matematycznego kinematyki i dynamiki kołowego robota mobilnego.

### 11.2.1 Ograniczenia fazowe

Niech układ  $X_0Y_0Z_0$  będzie podstawowym układem odniesienia, którego oś  $Z_0$  pokrywa się z kierunkiem wektora grawitacji. Rozważmy kołowy robot mobilny złożony z traktora ciągnącego  $p$  przyczep. Przyjmujemy, że każdy element składowy robota posiada jednorodne i недеformowalne koła<sup>†</sup> i że porusza się po płaszczyźnie  $X_0Y_0$  bez poślizgu. Po umieszczeniu lokalnego układu współrzędnych  $X_1Y_1Z_1$  na traktorze, do określenia położenia i orientacji traktora będziemy używać współrzędnych  $\boldsymbol{\xi} = (x, y, \theta)^T$ , gdzie  $x$  i  $y$  oznaczają położenie początku układu lokalnego w układzie  $X_0Y_0Z_0$ , zaś  $\theta$  jest kątem między osiami  $X_0$  i  $X_1$ .

W celu wprowadzenia ograniczeń fazowych dla kołowego robota mobilnego rozpatrzmy układ zawieszenia pojedynczego koła, złożony z  $s$  elementów połączonych przegubami obrotowymi (zobacz rysunek 11.1). Układ ten potraktujemy jako manipulator planarny o  $s$  przegubach z efektoorem w postaci koła, co pozwoli na wykorzystanie algorytmu Denavita-Harten-

\*Zauważmy, że wektor sterowań w układzie (11.8) jest inny niż w układzie (11.6).

<sup>†</sup>Oznacza to, że kontakt koła z podłożem ogranicza się do pojedynczego punktu.

berga opisanego w podrozdziale 2.3.1. W rezultacie, współrzędne jednorodne  $\mathbf{p} = (p_1, p_2, p_3, 1)^T$  pewnego punktu na obwodzie koła względem układu  $X_0Y_0Z_0$  można wyrazić wzorem

$$\mathbf{p} = \mathbf{A}_0^2(\boldsymbol{\xi}, \mathbf{l}, \boldsymbol{\alpha})\mathbf{p}_2, \quad (11.9)$$

gdzie

$\mathbf{p}_2 = (0, r \cos \phi, r \sin \phi, 1)^T$  — współrzędne jednorodne wybranego punktu na obwodzie koła w układzie lokalnym  $X_2Y_2Z_2$ †,

$r$  — promień koła,

$\boldsymbol{\xi} = (x, y, \theta)^T$  — położenie i orientacja traktora względem układu  $X_0Y_0Z_0$ ,

$\phi$  — kąt obrotu koła wokół osi  $X_2$  układu lokalnego  $X_2Y_2Z_2$ ,

$\mathbf{A}_0^2(\boldsymbol{\xi}, \mathbf{l}, \boldsymbol{\alpha}) = \mathbf{Trans}(X, x)\mathbf{Trans}(Y, y)\mathbf{Trans}(Z, z)\mathbf{Rot}(Z, \theta)$   
 $\left( \prod_{i=1}^s \mathbf{Rot}(Z, \alpha_i)\mathbf{Trans}(X, l_i) \right) \mathbf{Rot}(Z, \gamma),$

$(\mathbf{l}, \boldsymbol{\alpha})$  — charakterystyka geometryczna koła, czyli parametry określające położenie i orientację koła w układzie  $X_1Y_1Z_1$ ,

$\mathbf{l} = (l_1, l_2, \dots, l_s)^T, \boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_s, \gamma)^T$ .

Przyjmujemy, że kąt  $\gamma$  jest stały w czasie. W celu wyznaczenia warunków toczenia się koła bez poślizgu zróżniczkujemy zależność (11.9) względem czasu. Warunki te są związane z punktem B (w tym punkcie  $\phi = -\pi/2$ ) o współrzędnych  $\mathbf{b} = \mathbf{A}_0^2(\boldsymbol{\xi}, \mathbf{l}, \boldsymbol{\alpha})\mathbf{b}_2$ , w którym ma miejsce kontakt koła z podłożem i sprowadzają się do wymagania, aby prędkość punktu B względem układu  $X_0Y_0Z_0$  była równa  $\mathbf{0}$ , a więc  $\dot{\mathbf{b}} = \mathbf{0}$ . Zatem, założenie o toczeniu się koła bez poślizgu po płaszczyźnie poziomej pociąga za sobą następujące zależności

$$\frac{d}{dt}\mathbf{A}_0^2(\boldsymbol{\xi}, \mathbf{l}, \boldsymbol{\alpha})\mathbf{b}_2 + \mathbf{A}_0^2(\boldsymbol{\xi}, \mathbf{l}, \boldsymbol{\alpha})\dot{\mathbf{b}}_2 = \mathbf{0}.$$

Wyrażenie warunku braku poślizgu w układzie  $X_2Y_2Z_2$ ,

$$\dot{\mathbf{b}}_2 + (\mathbf{A}_0^2)^{-1}(\boldsymbol{\xi}, \mathbf{l}, \boldsymbol{\alpha})\frac{d}{dt}\mathbf{A}_0^2(\boldsymbol{\xi}, \mathbf{l}, \boldsymbol{\alpha})\mathbf{b}_2 = \mathbf{0},$$

pozwała zdekomponować toczenie się koła bez poślizgu na toczenie się bez

†Układ ten jest nieruchomo związany z osią koła w taki sposób, że płaszczyzna  $X_2Y_2$  jest równoległa do płaszczyzny  $X_0Y_0$ , a os  $X_2$  jest prostopadła do płaszczyzny koła.

poślizgu bocznego i wzdłużnego. Ostatecznie, biorąc pod uwagę położenie  $\phi = -\pi/2$  punktu kontaktu koła z podłożem, wyliczamy

$$\mathbf{b}_2 = \begin{pmatrix} 0 \\ r \cos \phi \\ r \sin \phi \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -r \\ 1 \end{pmatrix}, \quad \dot{\mathbf{b}}_2 = \begin{pmatrix} 0 \\ -r\dot{\phi} \sin \phi \\ r\dot{\phi} \cos \phi \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ r\dot{\phi} \\ 0 \\ 0 \end{pmatrix}.$$

Wymaganie toczenia się bez poślizgu przybiera zatem formę następujących ograniczeń fazowych

$$\begin{pmatrix} 0 \\ r\dot{\phi} \\ 0 \\ 0 \end{pmatrix} = -(\mathbf{A}_0^2)^{-1}(\boldsymbol{\xi}, \mathbf{l}, \boldsymbol{\alpha}) \frac{d}{dt} \mathbf{A}_0^2(\boldsymbol{\xi}, \mathbf{l}, \boldsymbol{\alpha}) \begin{pmatrix} 0 \\ 0 \\ -r \\ 1 \end{pmatrix}, \quad (11.10)$$

z których, po uwzględnieniu postaci macierzy  $\mathbf{A}_0^2(\boldsymbol{\xi}, \mathbf{l}, \boldsymbol{\alpha})$ , otrzymujemy równania opisujące:

- *toczenie się koła bez poślizgu bocznego*

$$\begin{aligned} & \left( \cos \bar{\alpha}_1, \sin \bar{\alpha}_1, \sum_{i=1}^s l_i \sin \bar{\alpha}_{i+1} \right) \mathbf{R}(-\theta) \dot{\boldsymbol{\xi}} + \dot{\alpha}_1 \left( \sum_{i=1}^s l_i \sin \bar{\alpha}_{i+1} \right) + \\ & + \dot{\alpha}_2 \left( \sum_{i=2}^s l_i \sin \bar{\alpha}_{i+1} \right) + \dots + \dot{\alpha}_s l_s \sin \gamma = 0, \quad (11.11) \end{aligned}$$

- *toczenie się koła bez poślizgu wzdłużnego*

$$\begin{aligned} & \left( -\sin \bar{\alpha}_1, \cos \bar{\alpha}_1, \sum_{i=1}^s l_i \cos \bar{\alpha}_{i+1} \right) \mathbf{R}(-\theta) \dot{\boldsymbol{\xi}} + \dot{\alpha}_1 \left( \sum_{i=1}^s l_i \cos \bar{\alpha}_{i+1} \right) + \\ & + \dot{\alpha}_2 \left( \sum_{i=2}^s l_i \cos \bar{\alpha}_{i+1} \right) + \dots + \dot{\alpha}_s l_s \cos \gamma + r\dot{\phi} = 0, \quad (11.12) \end{aligned}$$

- *toczenie się koła po płaszczyźnie*

$$\dot{z} = 0. \quad (11.13)$$

W równaniach (11.11)–(11.13)  $\bar{\alpha}_i = \gamma + \sum_{k=i}^s \alpha_k$ , a  $\mathbf{R}(-\theta) = \mathbf{R}(Z, -\theta)$  oznacza macierz obrotu o kąt  $-\theta$  wokół osi Z układu  $x_1 y_1 z_1$ .

Warunki braku poślizgu bocznego i wzdłużnego kół wyprowadzone przy pomocy formalizmu Denavita-Hartenberga umożliwiają wyznaczenie ogra-



niczeń fazowych dla prostych i złożonych kołowych robotów mobilnych poruszających się po płaszczyźnie poziomej.

### Przykład 11.2.1 (Jednokołowy robot mobilny)

Na podstawie analizy układów zawieszenia kół kołowych robotów mobilnych okazało się, że charakterystyka geometryczna koła postaci  $\mathbf{l} = (l_1, l_2)^T$  oraz  $\boldsymbol{\alpha} = (\alpha_1 = \text{const}, \alpha_2 = \text{var}, \gamma = \text{const})^T$  opisuje wszystkie spotykane w praktyce rozwiązania. Na tej podstawie, korzystając z ogólnych zależności (11.11)–(11.13) przedstawiających ograniczenia fazowe dla pojedynczego koła, otrzymujemy warunki:

- toczenia się koła bez poślizgu bocznego

$$(\cos \alpha, \sin \alpha, l_2 \sin \gamma + l_1 \sin(\alpha_1 + \gamma)) \mathbf{R}(-\theta) \dot{\boldsymbol{\xi}} + l_2 \dot{\alpha}_2 \sin \gamma = 0,$$

- toczenia się koła bez poślizgu wzdłużnego

$$(-\sin \alpha, \cos \alpha, l_2 \cos \gamma + l_1 \cos(\alpha_1 + \gamma)) \mathbf{R}(-\theta) \dot{\boldsymbol{\xi}} + l_2 \dot{\alpha}_2 \cos \gamma + r \dot{\phi} = 0,$$

gdzie  $\alpha = \alpha_1 + \alpha_2 + \gamma$ . Przedstawione w postaci Pfaffa ograniczenia te wyglądają następująco<sup>§</sup>:

$$\begin{bmatrix} \cos(\alpha + \theta) & \sin(\alpha + \theta) & l_2 \sin \gamma + l_1 \sin(\alpha_1 + \gamma) & l_2 \sin \gamma & 0 \\ -\sin(\alpha + \theta) & \cos(\alpha + \theta) & l_2 \cos \gamma + l_1 \cos(\alpha_1 + \gamma) & l_2 \cos \gamma & r \end{bmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\alpha}_2 \\ \dot{\phi} \end{pmatrix} = \mathbf{0}.$$

Korzystając z twierdzenia 2.2.1 nietrudno wykazać, że ograniczenia fazowe dla jednokołowego robota mobilnego o charakterystyce geometrycznej  $(\mathbf{l}, \boldsymbol{\alpha})$  są nieholonomiczne. ■

Dla kołowego robota mobilnego złożonego z traktora ciągnącego p przyczep wymaganie toczenia się wszystkich kół bez poślizgu prowadzi, po uwzględnieniu zależności (11.11) i (11.12), do ograniczeń fazowych postaci

<sup>§</sup>Przy podstawieniach  $l_1 = l_2 = 0$ ,  $\alpha_1 = \alpha_2 = 0$ ,  $\gamma = -\pi/2$  pierwsze z poniższych równań przechodzi w równanie (8.2).

$$\left\{ \begin{array}{l} \mathbf{A}_0(\mathbf{q}_0)\dot{\mathbf{q}}_0 = \mathbf{0} \\ \mathbf{A}_{10}(\mathbf{q}_0, \mathbf{q}_1)\dot{\mathbf{q}}_0 + \mathbf{A}_{11}(\mathbf{q}_1)\dot{\mathbf{q}}_1 = \mathbf{0} \\ \vdots \\ \sum_{i=0}^r \mathbf{A}_{ri}(\mathbf{q}_i, \mathbf{q}_{i+1}, \dots, \mathbf{q}_r)\dot{\mathbf{q}}_i = \mathbf{0} \\ \vdots \\ \sum_{i=0}^p \mathbf{A}_{pi}(\mathbf{q}_i, \mathbf{q}_{i+1}, \dots, \mathbf{q}_p)\dot{\mathbf{q}}_i = \mathbf{0}, \end{array} \right. \quad (11.14)$$

przy oznaczeniach

$\mathbf{q}_0$  — współrzędne uogólnione traktora,

$\mathbf{A}_0(\mathbf{q}_0)\dot{\mathbf{q}}_0 = \mathbf{0}$  — ograniczenia fazowe związane z traktorem,

$\mathbf{q}_i, i = 1, 2, \dots, p$  — współrzędne uogólnione  $i$ -tej przyczepy,

$\sum_{i=0}^r \mathbf{A}_{ri}(\mathbf{q}_i, \mathbf{q}_{i+1}, \dots, \mathbf{q}_r)\dot{\mathbf{q}}_i = \mathbf{0}$  — ograniczenia fazowe związane z  $r$ -tą przyczepą.

Nietrudno zauważyć, że równania ograniczeń fazowych (11.14) dla wieloczołowego kołowego robota mobilnego dają się zapisać w postaci Pfaffa (11.1).

## 11.2.2 Modele kinematyki prostych kołowych robotów mobilnych

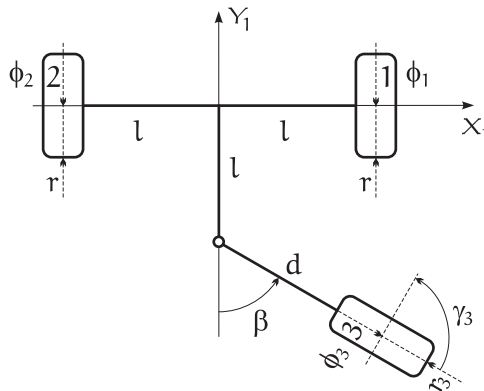
Przyjmijmy, że prosty kołowy robot mobilny (traktor) posiada  $k$  kół, z których każde ma promień  $r_i$  i jest opisane przez charakterystykę geometryczną  $(\mathbf{l}_i, \boldsymbol{\alpha}_i) = (l_{i1}, l_{i2}, \alpha_{i1} = \text{const}, \alpha_{i2}, \gamma_i = \text{const})^T, i = 1, 2, \dots, k$ . Warunki ruchu bez poślizgu postaci (11.11) i (11.12) prowadzą do następujących ograniczeń fazowych:

- ruch traktora bez poślizgu bocznego

$$\mathbf{C}_1(\boldsymbol{\alpha}_2)\mathbf{R}(-\theta) \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix}^T + \mathbf{C}_2\dot{\boldsymbol{\alpha}}_2 = \mathbf{0}, \quad (11.15)$$

gdzie macierze  $\mathbf{C}_1, \mathbf{C}_2$  są zdefiniowane jako

- $\mathbf{C}_{1i}$ -ty wiersz  $= (\cos(\alpha_{i1} + \alpha_{i2} + \gamma_i), \sin(\alpha_{i1} + \alpha_{i2} + \gamma_i), l_{i2} \sin \gamma_i + l_{i1} \sin(\alpha_{i2} + \gamma_i)), i = 1, 2, \dots, k$ ,
- $\mathbf{C}_2 = \text{diag}\{l_{i2} \sin \gamma_i\}, i = 1, 2, \dots, k$ ,



Rysunek 11.2 Schemat kinematyczny traktora z kołem swobodnym.

- ruch traktora bez poślizgu wzdłużnego

$$\mathbf{Z}_1(\boldsymbol{\alpha}_2)\mathbf{R}(-\theta) \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix}^T + \mathbf{Z}_2\boldsymbol{\alpha}_2 + \mathbf{Z}_3\dot{\boldsymbol{\phi}} = \mathbf{0}, \quad (11.16)$$

gdzie macierze  $\mathbf{Z}_1$ ,  $\mathbf{Z}_2$ ,  $\mathbf{Z}_3$  są zdefiniowane przez

- $\mathbf{Z}_{1i}$ -ty wiersz  $(\boldsymbol{\alpha}_2) = (-\sin(\alpha_{i1} + \alpha_{i2} + \gamma_i), \cos(\alpha_{i1} + \alpha_{i2} + \gamma_i), l_{i2} \cos \gamma_i + l_{i2} \cos(\alpha_{i2} + \gamma_i))$ ,  $i = 1, 2, \dots, k$ ,
- $\mathbf{Z}_2 = \text{diag}\{l_{i2} \cos \gamma_i\}$ ,  $i = 1, 2, \dots, k$ ,
- $\mathbf{Z}_3 = \text{diag}\{r_i\}$ ,  $i = 1, 2, \dots, k$ .

W zależnościach (11.15) i (11.16) wektor  $\boldsymbol{\alpha}_2 = (\alpha_{12}, \alpha_{22}, \dots, \alpha_{k2})^T$ , a  $i$ -ty wiersz jest równaniem ograniczeń dla  $i$ -tego koła rozpatrywanego kołowego robota mobilnego.

### Przykład 11.2.2 (Traktor trójkołowy)

Rozpatrzmy przypadek traktora, którego układ jezdny składa się z dwóch sztywnych kół tylnych oraz z koła swobodnego (wolnego) umieszczonego z przodu (zobacz rysunek 11.2). Lokalny układ współrzędnych  $X_1 Y_1 Z_1$  związany z poruszającym się robotem jest ulokowany w środku odcinka łączącego koła tylne (1 i 2). Orientacja  $\beta$  koła przedniego (3) względem lokalnego układu odniesienia ulega zmianie podczas ruchu traktora, podczas gdy orientacje kół tylnych pozostają stałe. Traktor trójkołowy jest wprowadzany w ruch przy pomocy dwóch silników napędzających koła tylne. Ograniczenia fazowe dla traktora z kołem swobodnym wyznaczamy w następujący sposób.

**Krok 1.** Określamy charakterystykę geometryczną kół traktora:

koło 1 —  $l_{11} = l$ ,  $\alpha_{11} = 0$ ,  $l_{12} = 0$ ,  $\alpha_{12} = 0$ ,  $\gamma_1 = 0$ ,

koło 2 —  $l_{21} = l$ ,  $\alpha_{21} = \pi$ ,  $l_{22} = 0$ ,  $\alpha_{22} = 0$ ,  $\gamma_2 = 0$ ,

koło 3 —  $l_{31} = l$ ,  $\alpha_{31} = 3/2\pi$ ,  $l_{32} = d$ ,  $\alpha_{32} = \beta$ ,  $\gamma_3 = \pi/2$ .

**Krok 2.** W oparciu o charakterystykę geometryczną kół oraz ogólne zależności (11.15) i (11.16) wyznaczamy ograniczenia fazowe:

- dla ruchu robota bez poślizgu bocznego

$$\begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \\ \cos \beta & \sin \beta & d + l \cos \beta \end{bmatrix} \mathbf{R}(-\theta) \dot{\boldsymbol{\xi}} + \begin{pmatrix} 0 \\ 0 \\ d \end{pmatrix} \dot{\beta} = \\ = \mathbf{C}_1(\beta) \mathbf{R}(-\theta) \dot{\boldsymbol{\xi}} + \mathbf{C}_2 \dot{\boldsymbol{\alpha}}_2 = \mathbf{0}, \quad (11.17)$$

- dla ruchu robota bez poślizgu wzdłużnego

$$\begin{bmatrix} 0 & 1 & l \\ 0 & -1 & l \\ -\sin \beta \cos \beta & -l \sin \beta \end{bmatrix} \mathbf{R}(-\theta) \dot{\boldsymbol{\xi}} + \begin{bmatrix} r & 0 & 0 \\ 0 & r & 0 \\ 0 & 0 & r_3 \end{bmatrix} \begin{pmatrix} \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{pmatrix} = \mathbf{0}. \quad (11.18)$$

Specyficzną postać ograniczeń kinematycznych (11.15) i (11.16) wykorzystuje się w algorytmie wyznaczania modelu kinematyki prostych kołowych robotów mobilnych. Algorytm ten składa się z następujących kroków.

**Krok 1.** Jeżeli macierz  $\mathbf{C}_2$  jest nieosobliwa (tzn. dla każdego koła robota mobilnego zachodzi warunek  $l_{2i} \sin \gamma_i \neq 0$ ), to model kinematyki robota przyjmujemy w postaci

$$\dot{\mathbf{q}}_0 = \begin{pmatrix} \dot{\boldsymbol{\xi}} \\ \dot{\boldsymbol{\alpha}}_2 \\ \dot{\boldsymbol{\phi}} \end{pmatrix} = \begin{bmatrix} \mathbf{R}(\theta) \\ -\mathbf{C}_2^{-1} \mathbf{C}_1(\boldsymbol{\alpha}_2) \\ -\mathbf{Z}_3^{-1} (\mathbf{Z}_1(\boldsymbol{\alpha}_2) - \mathbf{Z}_2 \mathbf{C}_2^{-1} \mathbf{C}_1(\boldsymbol{\alpha}_2)) \end{bmatrix} \mathbf{u} = \mathbf{G}_0(\mathbf{q}_0) \mathbf{u}, \quad (11.19)$$

gdzie  $\mathbf{u}$  jest wektorem sterowań. W przypadku przeciwnym przechodzimy do kroku 2.

**Krok 2.** Dzielimy ograniczenia związane z ruchem robota bez poślizgu bocznego (11.15) na dwie grupy

$$\begin{cases} \mathbf{C}_{1v}(\boldsymbol{\alpha}_{2v}) \mathbf{R}(-\theta) \dot{\boldsymbol{\xi}} + \mathbf{C}_{2v} \dot{\boldsymbol{\alpha}}_{2v} = \mathbf{0} \\ \mathbf{C}_{1u}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q}) \mathbf{R}(-\theta) \dot{\boldsymbol{\xi}} = \mathbf{0}, \end{cases} \quad (11.20)$$

gdzie

$\mathbf{C}_{2v}$  — diagonalna macierz nieosobliwa,

$$\boldsymbol{\alpha}_2 = (\boldsymbol{\alpha}_{2v}, \boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q})^T,$$

$\boldsymbol{\alpha}_{2v}$  — wektor zawierający zmienne w czasie współrzędne wektora  $\boldsymbol{\alpha}_2$ , dla których zachodzi  $l_{2i} \sin \gamma_i \neq 0$ ,

$\boldsymbol{\alpha}_{2u}$  — wektor zawierający zmienne w czasie współrzędne wektora  $\boldsymbol{\alpha}_2$ , dla których zachodzi  $l_{2i} \sin \gamma_i = 0$ ,

$\boldsymbol{\alpha}_{2q}$  — wektor złożony ze stałych w czasie współrzędnych wektora  $\boldsymbol{\alpha}_2$ .

**Krok 3.** Szukamy macierzy  $\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q})$ , której kolumny rozpinają przestrzeń zerową macierzy (jądro)  $\mathbf{C}_{1u}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q})$

$$\mathbf{C}_{1u}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q})\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q}) = \mathbf{0}. \quad (11.21)$$

Przy wyznaczaniu postaci macierzy  $\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q})$  korzystamy z następującego twierdzenia.

**Twierdzenie 11.2.1** *Zdefiniujmy stopień mobilności  $\delta_m$  kołowego robota mobilnego równy wymiarowi dystrybucji rozpiętej przez kolumny macierzy  $\mathbf{R}(\theta)\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q})$  oraz stopień sterowalności  $\delta_s$  jako liczbę sterowanych kół robota, których orientacja może być ustalona w sposób niezależny, tak że istnieje chwilowy środek obrotu robota. Istnieje 5 podstawowych typów kołowych robotów mobilnych poruszających się po płaszczyźnie, z których każdy może być w pełni scharakteryzowany przez parę liczb  $(\delta_m, \delta_s)$ . Dla każdego typu robota można znaleźć taki punkt (w obrębie układu jezdnego lub nadwozia robota), zwany punktem charakterystycznym, że umieszczając w nim odpowiednio zorientowany lokalny układ odniesienia otrzymamy jedną z następujących postaci macierzy  $\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q})$ :*

- typ (3, 0)

$$\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q}) = \mathbb{I}_3, \quad (11.22)$$

- typ (2, 0)

$$\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q}) = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad (11.23)$$

- typ (2, 1)

$$\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q}) = \begin{bmatrix} -\sin \alpha_{2u1} & 0 \\ \cos \alpha_{2u1} & 0 \\ 0 & 1 \end{bmatrix}, \quad (11.24)$$

- typ (1, 1)

$$\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q}) = \begin{bmatrix} 0 \\ l \sin \alpha_{2u1} \\ \cos \alpha_{2u1} \end{bmatrix}, \quad (11.25)$$

- typ (1, 2)

$$\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q}) = \begin{bmatrix} -2l \sin \alpha_{2u1} \sin \alpha_{2u2} \\ l \sin (\alpha_{2u1} + \alpha_{2u2}) \\ \sin (\alpha_{2u2} - \alpha_{2u1}) \end{bmatrix}. \quad (11.26)$$

Parametr  $l$  oznacza odległość punktu charakterystycznego robota do pewnego punktu związanego z układem napędowym.

Odpowiedni wybór położenia lokalnego układu odniesienia związanego z punktem charakterystycznym kołowego robota mobilnego umożliwia opis każdego z kół przy pomocy charakterystyki geometrycznej postaci  $(l, \boldsymbol{\alpha})$ .

Z drugiego z równań (11.20) i równania (11.21) wynika zależność

$$\dot{\boldsymbol{\xi}} = \mathbf{R}(\boldsymbol{\theta})\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q})\mathbf{u}_1, \quad (11.27)$$

gdzie  $\mathbf{u}_1$  jest wektorem sterowań. Praktyczny sposób wyznaczania macierzy  $\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q})$  polega na sprawdzeniu, która z macierzy (11.23)–(11.26) spełnia warunek opisany przez (11.21). Jeżeli przy poprawnym wyborze punktu charakterystycznego żadna z proponowanych postaci macierzy  $\mathbf{P}(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q})$  nie spełnia tego warunku, oznacza to, że ruch rozpatrywanego układu może się odbywać przy dodatkowych ograniczeniach konfiguracyjnych\*, lub że robot mobilny może poruszać się po płaszczyźnie  $X_0Y_0$  wzdłuż pewnego kierunku określonego przez warunki początkowe. W przypadku, gdy jedna z macierzy postaci (11.23)–(11.26) spełnia warunek (11.21), przechodzimy do kroku 4.

\*Związki otrzymywane z warunku  $\det \mathbf{C}_1(\boldsymbol{\alpha}_{2u}, \boldsymbol{\alpha}_{2q}) = 0$ .

**Krok 4.** Z pierwszej z zależności (11.20) oraz z (11.27) obliczamy

$$\dot{\alpha}_{2v} = -\mathbf{C}_{2v}^{-1} \mathbf{C}_1(\alpha_{2v}) \mathbf{R}(-\theta) \dot{\xi} = -\mathbf{C}_2^{-1} \mathbf{C}_1(\alpha_{2v}) \mathbf{P}(\alpha_{2u}, \alpha_{2q}) \mathbf{u}_1. \quad (11.28)$$

**Krok 5.** Wykorzystując dekompozycję wektora  $\alpha_2$ , ograniczenia związane z ruchem robota bez poślizgu wzdłużnego dane przez (11.16) możemy przedstawić w formie

$$\begin{cases} \mathbf{Z}_{1v}(\alpha_{2v}) \mathbf{R}(-\theta) \dot{\xi} + \mathbf{Z}_{2v}(\alpha_{2v}) \dot{\alpha}_{2v} + \mathbf{Z}_{3v} \dot{\phi}_v = \mathbf{0} \\ \mathbf{Z}_{1u}(\alpha_{2u}) \mathbf{R}(-\theta) \dot{\xi} + \mathbf{Z}_{2u}(\alpha_{2u}) \dot{\alpha}_{2u} + \mathbf{Z}_{3u} \dot{\phi}_u = \mathbf{0} \\ \mathbf{Z}_{1q}(\alpha_{2q}) \mathbf{R}(-\theta) \dot{\xi} + \mathbf{Z}_{3q} \dot{\phi}_q = \mathbf{0}, \end{cases}$$

gdzie  $\dot{\phi}_v$ ,  $\dot{\phi}_u$  i  $\dot{\phi}_q$  oznaczają prędkości obrotu kół robota mobilnego określone stosownie do podziału wektora  $\alpha_2$ . Z kolei, z uwagi na nieosobliwość macierzy  $\mathbf{Z}_{3u}$ ,  $\mathbf{Z}_{3v}$  i  $\mathbf{Z}_{3q}$  oraz równania (11.27) i (11.28), otrzymujemy

$$\begin{cases} \dot{\phi}_v = -\mathbf{Z}_{3v}^{-1} (\mathbf{Z}_{1v}(\alpha_{2v}) - \mathbf{Z}_{2v}(\alpha_2) \mathbf{C}_{2v}^{-1} \mathbf{C}_{1v}(\alpha_{2v})) \mathbf{P}(\alpha_{2u}, \alpha_{2q}) \mathbf{u}_1 \\ \dot{\phi}_u = -\mathbf{Z}_{3u}^{-1} (\mathbf{Z}_{1u}(\alpha_{2u}) \mathbf{P}(\alpha_{2u}, \alpha_{2q}) \mathbf{u}_1 + \mathbf{Z}_{2u}(\alpha_{2u}) \mathbf{u}_2) \\ \dot{\phi}_q = -\mathbf{Z}_{3q}^{-1} \mathbf{Z}_{1q}(\alpha_{2q}) \mathbf{P}(\alpha_{2u}, \alpha_{2q}) \mathbf{u}_1 \\ \dot{\alpha}_{2u} = \mathbf{u}_2. \end{cases} \quad (11.29)$$

Reasumując, zależności (11.27), (11.28) i (11.29) wyznaczają następujący model kinematyki robota

$$\dot{\mathbf{q}}_0 = \mathbf{G}_0(\mathbf{q}_0) \mathbf{u}, \quad (11.30)$$

gdzie

$$\mathbf{G}_0(\mathbf{q}_0) = \begin{bmatrix} \mathbf{P}(\alpha_{2u}, \alpha_{2q}) & \mathbf{0} \\ -\mathbf{C}_{2v} \mathbf{C}_{1v}(\alpha_{2v}) \mathbf{P}(\alpha_{2u}, \alpha_{2q}) & \mathbf{0} \\ \mathbf{0} & \mathbb{I}_u \\ \mathbf{S}_{\phi v}(\alpha_{2v}) & \mathbf{0} \\ -\mathbf{Z}_{3u}^{-1} \mathbf{Z}_{1u}(\alpha_{2u}) \mathbf{P}(\alpha_{2u}, \alpha_{2q}) & \mathbf{Z}_{2u}(\alpha_{2u}) \\ -\mathbf{Z}_{3q}^{-1} \mathbf{Z}_{1q}(\alpha_{2q}) \mathbf{P}(\alpha_{2u}, \alpha_{2q}) & \mathbf{0} \end{bmatrix},$$

oraz  $\mathbf{q}_0 = (\dot{\xi}, \dot{\alpha}_{2v}, \dot{\alpha}_{2u}, \dot{\phi}_v, \dot{\phi}_u, \dot{\phi}_q)^\top$ ,  $\mathbf{u} = (\mathbf{u}_1^\top, \mathbf{u}_2^\top)^\top$ ,  $\mathbf{S}_{\phi v}(\alpha_{2v}) = -\mathbf{Z}_{3v}^{-1} (\mathbf{Z}_{1v}(\alpha_{2v}) - \mathbf{Z}_{2v}(\alpha_2) \mathbf{C}_{2v}^{-1} \mathbf{C}_{1v}(\alpha_{2v})) \mathbf{P}(\alpha_{2u}, \alpha_{2q})$ .

Przedstawiony algorytm wyznaczania modelu kinematyki prostych kołowych robotów mobilnych generuje model w postaci (11.19) lub (11.30), zależnie od osobliwości lub nieosobliwości macierzy  $\mathbf{C}_2$ .

### Przykład 11.2.3 (Traktor trójkołowy)

Rozważmy teraz zadanie określenia modelu kinematyki traktora trójkołowego z kołem swobodnym. Opis robota oraz wyprowadzenie ograniczeń fazowych przedstawiliśmy w przykładzie 11.2.2. Macierz  $\mathbf{C}_2$  w równaniu (11.17) nie jest pełnego rzędu, zatem dokonujemy dekompozycji ograniczeń fazowych związanych z ruchem układu bez poślizgu bocznego na dwie części:

$$\begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \mathbf{R}(-\theta)\dot{\xi} = \mathbf{C}_{1u}\mathbf{R}(-\theta)\dot{\xi} = \mathbf{0}, \quad (11.31)$$

$$(\cos \beta, \sin \beta, d + l \cos \beta) \mathbf{R}(-\theta)\dot{\xi} + \dot{\beta}d = 0. \quad (11.32)$$

Korzystając z twierdzenia 11.2.1, szukamy macierzy  $\mathbf{P}$ , której kolumny rozpinają przestrzeń  $\text{Ker } \mathbf{C}_{1u}$ , tzn. takiej, że

$$\mathbf{C}_{1u}\mathbf{P} = \mathbf{0}.$$

Nietrudno zauważyć, że rozpatrywany traktor jest typu (2, 0), a zatem macierz  $\mathbf{P}$  ma postać

$$\mathbf{P} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Wnioskujemy stąd, że

$$\dot{\xi} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \mathbf{R}(\theta)\mathbf{P}\mathbf{u} = \begin{pmatrix} -u_1 \sin \theta \\ u_1 \cos \theta \\ u_2 \end{pmatrix}, \quad (11.33)$$

gdzie  $u_1$  jest prędkością liniową traktora z kołem swobodnym w układzie podstawowym, natomiast  $u_2$  prędkością zmian orientacji traktora. Z zależności (11.32) wyznaczamy zmienną  $\dot{\beta}$

$$\dot{\beta} = -u_1 \frac{1}{d} \sin \beta - u_2 \left( 1 + \frac{l}{d} \cos \beta \right), \quad (11.34)$$

natomiast z warunków na ruch traktora bez poślizgu wzdłużnego (11.18)



zmienne  $\dot{\phi}_1$ ,  $\dot{\phi}_2$  i  $\dot{\phi}_3$

$$\begin{cases} \dot{\phi}_1 = -u_1 \frac{1}{r} - u_2 \frac{l}{r} \\ \dot{\phi}_2 = u_1 \frac{1}{r} - u_2 \frac{l}{r} \\ \dot{\phi}_3 = -u_1 \frac{1}{r_3} \cos \beta - u_2 \frac{l}{r_3} \sin \beta. \end{cases} \quad (11.35)$$

Ostatecznie, korzystając z równań (11.33), (11.34) i (11.35), otrzymujemy model kinematyki traktora z kołem swobodnym w postaci

$$\dot{\mathbf{q}} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\beta} \\ \dot{\phi}_1 \\ \dot{\phi}_2 \\ \dot{\phi}_3 \end{pmatrix} = \begin{bmatrix} -\sin \theta & 0 \\ \cos \theta & 0 \\ 0 & 1 \\ -\frac{1}{d} \sin \beta & -(1 + \frac{1}{d} \cos \beta) \\ -\frac{1}{r} & -\frac{1}{r} \\ \frac{1}{r} & -\frac{1}{r} \\ -\frac{1}{r_3} \cos \beta & \frac{1}{r_3} \sin \beta \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \mathbf{G}(\beta, \theta) \mathbf{u}. \quad (11.36)$$

■

#### Przykład 11.2.4 (Robot mobilny Ulisses)

Przykładem traktora trójkołowego z kołem swobodnym jest kołowy robot mobilny Ulisses<sup>†</sup>. Model kinematyki robota Ulisses wynika z modelu (11.36) (po pominięciu współrzędnych  $\beta$  i  $\phi_3$ , na co pozwalają małe rozmiary koła swobodnego) i przyjmuje postać

$$\dot{\mathbf{q}} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi}_1 \\ \dot{\phi}_2 \end{pmatrix} = \begin{bmatrix} -\sin \theta & 0 \\ \cos \theta & 0 \\ 0 & 1 \\ -\frac{1}{r} & -\frac{1}{r} \\ \frac{1}{r} & -\frac{1}{r} \end{bmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \mathbf{G}(\theta) \mathbf{u}. \quad (11.37)$$

Parametry geometryczne występujące w przedstawionym modelu wynoszą:  $r = 0.05[\text{m}]$ ,  $l = 0.23[\text{m}]$ . ■

<sup>†</sup>Robot mobilny Ulisses został zbudowany w Zakładzie Podstaw Cybernetyki i Robotyki Instytutu Cybernetyki Technicznej Politechniki Wrocławskiej przez dra A. Wołczowskiego.

### 11.2.3 Modele kinematyki złożonych kołowych robotów mobilnych

Przedstawiony w poprzednim podrozdziale algorytm tworzenia modeli kinematyki prostych kołowych robotów mobilnych może być uogólniony na przypadek złożonych kołowych robotów mobilnych. Punktem wyjścia jest w tym przypadku ogólna postać ograniczeń kinematycznych (11.14). Proponowane rozwiązanie zadania wyznaczenia modeli kinematyki złożonych kołowych robotów mobilnych wykorzystuje specyficzną postać ograniczeń fazowych i jest oparte na spostrzeżeniu, że w przypadku ogólnym (złożony układ jezdny) model kinematyki daje się zapisać w postaci

$$\dot{\mathbf{q}} = \begin{pmatrix} \dot{\mathbf{q}}_0 \\ \dot{\mathbf{q}}_1 \\ \vdots \\ \dot{\mathbf{q}}_p \end{pmatrix} = \mathbf{G}(\mathbf{q})\mathbf{u} = \begin{bmatrix} \mathbf{G}_0(\mathbf{q}_0) \\ \mathbf{G}_1(\mathbf{q}_0, \mathbf{q}_1) \\ \vdots \\ \mathbf{G}_p(\mathbf{q}_0, \mathbf{q}_1, \dots, \mathbf{q}_p) \end{bmatrix} \mathbf{u}. \quad (11.38)$$

W rezultacie, na mocy warunku  $\mathbf{A}(\mathbf{q})\mathbf{G}(\mathbf{q}) = \mathbf{0}$ , z (11.14) otrzymujemy następujące zależności

$$\begin{cases} \mathbf{A}_0(\mathbf{q}_0)\mathbf{G}_0(\mathbf{q}_0) = \mathbf{0} \\ \mathbf{A}_{10}(\mathbf{q}_0, \mathbf{q}_1)\mathbf{G}_0(\mathbf{q}_0) + \mathbf{A}_{11}(\mathbf{q}_1)\mathbf{G}_1(\mathbf{q}_0, \mathbf{q}_1) = \mathbf{0} \\ \vdots \\ \sum_{i=0}^p \mathbf{A}_{pi}(\mathbf{q}_i, \mathbf{q}_{i+1}, \dots, \mathbf{q}_p)\mathbf{G}_i(\mathbf{q}_0, \mathbf{q}_1, \dots, \mathbf{q}_i) = \mathbf{0}. \end{cases} \quad (11.39)$$

Struktura równania (11.39) wskazuje na możliwość dekompozycji problemu wyznaczania macierzy  $\mathbf{G}(\mathbf{q})$  na  $p + 1$  kolejnych zadań polegających na obliczeniu podmacierzy  $\mathbf{G}_i(\mathbf{q}_0, \mathbf{q}_1, \dots, \mathbf{q}_i)$  dla  $i = 0, 1, 2, \dots, p$ , co w efekcie pozwala skonstruować algorytm wyznaczania modeli kinematyki dla złożonych kołowych robotów mobilnych. Algorytm ten składa się z następujących kroków.

**Krok 1.** Obliczamy macierz  $\mathbf{G}_0(\mathbf{q}_0)$  spełniającą zależność

$$\mathbf{A}(\mathbf{q}_0)\mathbf{G}_0(\mathbf{q}_0) = \mathbf{0}.$$

W tym celu możemy wykorzystać algorytm zamieszczony w podrozdziale 11.2.2, na stronie 360.

**Krok 2.** Uwzględniając pozostałe warunki (11.39), z układu równań macierzowych (11.39) wyznaczamy podmacierze  $\mathbf{G}_i(\mathbf{q}_0, \mathbf{q}_1, \dots, \mathbf{q}_i)$ ,  $i = 1, 2, \dots, p$  i otrzymujemy

$$\begin{aligned} \mathbf{G}_i(\mathbf{q}_0, \dots, \mathbf{q}_i) &= \\ &= -\mathbf{A}_{ii}^{-1}(\mathbf{q}_i) \sum_{k=0}^{i-1} \mathbf{A}_{ik}(\mathbf{q}_k, \dots, \mathbf{q}_i) \mathbf{G}_i(\mathbf{q}_0, \dots, \mathbf{q}_k). \end{aligned}$$

Przedstawiony algorytm wyznaczania modelu kinematyki złożonych kołowych robotów mobilnych działa poprawnie przy założeniu nieosobliwości macierzy  $\mathbf{A}_{ii}(\mathbf{q}_i)$ . Warunek ten nie jest spełniony dla robotów zdegenerowanych, czyli takich, w których obecność przyczep jest przyczyną zmniejszenia stopnia mobilności.

#### 11.2.4 Modele dynamiki

Po szczegółowym omówieniu zagadnień dotyczących modelowania kinematyki kołowych robotów mobilnych zajmiemy się wyznaczeniem zależności opisujących ich dynamikę. Ogólne zasady obowiązujące przy modelowaniu dynamiki układów robotycznych z więzami nieholonomicznymi zostały opisane w podrozdziale 11.1. Równania dynamiki uwzględniające wpływ sterowań (uogólnionych sił zewnętrznych wymuszających ruch układu) mają postać (11.6). Zadanie modelowania dynamiki nieholonomicznych układów robotycznych wymaga znajomości ich energii kinetycznej i potencjalnej. Jak łatwo zauważyć, energia potencjalna kołowego robota mobilnego o jednorodnych kołach, poruszającego się po płaszczyźnie prostopadłej do wektora grawitacji jest stała w czasie, a wobec tego nie wpływa na równania dynamiki robota. Energia kinetyczna kołowego robota mobilnego jest sumą energii układu jezdnego i nadwozia traktora, przyczep oraz energii obracających się kół. Połączenie klasycznej metodologii wyznaczania energii kinetycznej ciał sztywnych oraz reguł obliczania modeli kinematyki umożliwia wyprowadzenie kompletnych modeli matematycznych prostych i złożonych kołowych robotów mobilnych. Sposób tworzenia modelu zilustrujemy na prostym przykładzie.

##### Przykład 11.2.5 (Traktor trójkołowy)

Rozpatrzmy zadanie wyznaczenia modelu dynamiki traktora z kołem swobodnym zaniedbując, ze względu na małe wymiary geometryczne, wpływ koła swobodnego. Z przykładów 11.2.2 i 11.2.3 wynika, że wektor współ-

rzędnych uogólnionych opisujących zachowanie traktora jest postaci

$$\mathbf{q} = (x, y, \theta, \phi_1, \phi_2)^T, \quad (11.40)$$

natomiast macierz

$$\mathbf{G}(\mathbf{q}) = \begin{bmatrix} -\sin \theta & 0 \\ \cos \theta & 0 \\ 0 & 1 \\ -\frac{1}{r} & -\frac{1}{r} \\ \frac{1}{r} & -\frac{1}{r} \end{bmatrix}. \quad (11.41)$$

Przy założeniu, że koła 1 i 2 są napędzane niezależnie, macierz sterowań przybiera formę

$$\mathbf{B}(\mathbf{q}) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T. \quad (11.42)$$

Energię kinetyczną traktora, przy pominięciu energii koła swobodnego, możemy zapisać jako

$$E_k(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} m_c (\dot{x}^2 + \dot{y}^2) + \frac{1}{2} I_z \dot{\theta}^2 + \frac{1}{2} I_k (\dot{\phi}_1^2 + \dot{\phi}_2^2) = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{Q} \dot{\mathbf{q}}, \quad (11.43)$$

gdzie  $\mathbf{Q}(\mathbf{q}) = \text{diag}\{m_c, m_c, I_z, I_k, I_k\}$ ,  $m_c$  użyto do oznaczenia całkowitej masy traktora,  $I_z$  — całkowitego momentu bezwładności traktora względem osi  $Z$  układu  $X_1 Y_1 Z_1$ , a  $I_k$  jest momentem bezwładności koła napędzanego względem osi  $X$  odpowiedniego układu lokalnego\*. W oparciu o zależności (11.6), (11.40) oraz (11.43) wyznaczamy model dynamiki traktora

$$\begin{cases} \dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\boldsymbol{\eta} \\ \mathbf{G}^T(\mathbf{q})\mathbf{Q}(\mathbf{q})\mathbf{G}(\mathbf{q})\dot{\boldsymbol{\eta}} = \mathbf{G}^T(\mathbf{q})\mathbf{B}(\mathbf{q})\mathbf{u} \end{cases} \quad (11.44)$$

w którym  $\mathbf{G}^T(\mathbf{q})\mathbf{Q}(\mathbf{q})\mathbf{G}(\mathbf{q}) = \begin{bmatrix} m_c + 2I_k \frac{1}{r^2} & 0 \\ 0 & I_z + 2I_k \frac{1}{r^2} \end{bmatrix}$ ,  $\mathbf{G}^T(\mathbf{q})\mathbf{B}(\mathbf{q}) = \begin{bmatrix} -\frac{1}{r} & \frac{1}{r} \\ -\frac{1}{r} & -\frac{1}{r} \end{bmatrix}$ ,  $\mathbf{u} = (u_1, u_1)^T$ . ■

### Przykład 11.2.6 (Robot mobilny Ulisses)

Ponieważ kołowy robot mobilny Ulisses ma strukturę traktora trójkołowego z kołem swobodnym o pomijalnej charakterystyce geometrycznej, model jego dynamiki jest opisany równaniami (11.44). Parametry geometryczne robota podano w przykładzie 11.2.4, zaś parametry dynamiki przyjmują wartości:  $m_c = 12.8[\text{kg}]$ ,  $I_z = 0.089[\text{kg} \cdot \text{m}^2]$ ,  $I_k = 2.5 \times 10^{-5}[\text{kg} \cdot \text{m}^2]$ . ■

\*Przy założeniu, że koła 1 i 2 są identyczne.

## 11.3 Algorytmy sterowania kołowych robotów mobilnych

Przedstawimy teraz wybrane algorytmy sterowania kołowych robotów mobilnych lub, ogólniej, układów robotycznych z więzami nieholonomicznymi. Z punktu widzenia sterowania nieholonomicznych układów mechanicznych istotną rzeczą jest określenie specyficznych własności strukturalnych tych układów, które odróżniają je od układów holonomicznych. Temu celowi służą następujące twierdzenia.

**Twierdzenie 11.3.1** *Jeżeli macierz  $\mathbf{G}^T(\mathbf{q})\mathbf{B}(\mathbf{q})$  w równaniach dynamiki (11.6) jest nieosobliwa, to istnieje statyczne sprzężenie zwrotne*

$$\mathbf{u}(\boldsymbol{\eta}, \mathbf{q}, \mathbf{v}) : \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^m \longrightarrow \mathbb{R}^m$$

określone równaniem

$$\begin{aligned} \mathbf{G}^T(\mathbf{q})\mathbf{Q}(\mathbf{q})\mathbf{G}(\mathbf{q})\mathbf{v} + \mathbf{G}^T(\mathbf{q}) \left( \mathbf{C}(\mathbf{q}, \mathbf{G}(\mathbf{q})\boldsymbol{\eta})\mathbf{G}(\mathbf{q}) + \mathbf{Q}(\mathbf{q})\dot{\mathbf{G}}(\mathbf{q}) \right) \boldsymbol{\eta} + \\ + \mathbf{G}^T(\mathbf{q})\mathbf{D}(\mathbf{q}) = \mathbf{G}^T(\mathbf{q})\mathbf{B}(\mathbf{q})\mathbf{u}, \end{aligned}$$

gdzie  $\mathbf{v}$  oznacza  $m$ -wymiarowy wektor nowych sterowań, które przekształca układ (11.6) do postaci normalnej

$$\begin{cases} \dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\boldsymbol{\eta} \\ \dot{\boldsymbol{\eta}} = \mathbf{v}. \end{cases} \quad (11.45)$$

**Twierdzenie 11.3.2** *Przy założeniu, że ograniczenia fazowe są nieholonomiczne oraz  $\boldsymbol{\eta}$  jest wektorem sterowań, bezdrowy układ sterowania opisany przez pierwsze z równań (11.45) jest sterowalny\*.*

**Twierdzenie 11.3.3** *Układ sterowania opisany równaniami (11.45) posiada własność osiągalności z każdego punktu postaci  $(\mathbf{q}, \boldsymbol{\eta})$  (tzn. zbiór stanów osiągalnych z dowolnego punktu ma niepuste wnętrze).*

**Twierdzenie 11.3.4** *Punkt równowagi  $(\mathbf{q}_e, \mathbf{0})$  układu (11.45) nie jest stabilizowalny przez statyczne sprzężenie zwrotne od stanu.*

**Twierdzenie 11.3.5** *Układ sterowania (11.45) nie jest linearyzowalny wokół żadnego punktu  $(\mathbf{q}, \boldsymbol{\eta})$  przez statyczne sprzężenie zwrotne od stanu.*

\*W istocie, lokalnie sterowalny w krótkim czasie.

**Twierdzenie 11.3.6** *Układ sterowania (11.45) nie jest linearyzowalny dynamicznie w otoczeniu punktu równowagi  $(\mathbf{q}_e, \mathbf{0})$ .*

Przyjęta w dalszej części rozdziału kolejność opisywania algorytmów sterowania zmierza w kierunku wzrastającego stopnia nieznanomości modelu dynamiki. W podrozdziałach 11.3.1–11.3.4 zakładamy pełną znajomość modelu dynamiki, w podrozdziale 11.3.5 dopuszczamy nieznanomość parametryczną modelu, natomiast w podrozdziale 11.3.6 dopuszczamy strukturalną nieznanomość modelu dynamiki kołowego robota mobilnego.

### 11.3.1 Algorytm Corona-Pometa

Konsekwencją twierdzenia 11.3.4 jest to, że metody wykorzystujące gładkie statyczne sprzężenie zwrotne od stanu nie nadają się do rozwiązania zadania stabilizacji modelu postaci (11.45). Jednakże, przy spełnieniu pewnych warunków przez układ (11.45), zadanie stabilizacji może być rozwiązane przy użyciu zmiennego w czasie statycznego sprzężenia zwrotnego od stanu. Mówi o tym następujące twierdzenie.

**Twierdzenie 11.3.7** *Rozważmy układ sterowania postaci*

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u} = \sum_{k=1}^m \mathbf{g}_k(\mathbf{x})u_k, \quad (11.46)$$

gdzie  $\mathbf{q} \in \mathbb{R}^n$ ,  $\mathbf{u} \in \mathbb{R}^m$ . Załóżmy, że układ (11.46) spełnia warunki:

1. *Wymiar przestrzeni*

$$\dim \operatorname{span}_{\mathbb{C}^\infty} \left\{ \operatorname{ad}_{\mathbf{g}_i}^j \mathbf{g}_k \mid k = 2, \dots, m, j \geq 0 \right\}(\mathbf{q}) = n,$$

dla  $\mathbf{q}$  leżących w pewnym otoczeniu punktu  $\mathbf{0} \in \mathbb{R}^{n*}$ .

2. *Pole  $\mathbf{g}_1(\mathbf{q})$  można wyprostować w otoczeniu  $\mathbf{0} \in \mathbb{R}^{n\dagger}$ .*
3. *Jest określona pewna funkcja  $V(t, \mathbf{q})$  postaci*

$$V(t, \mathbf{q}) = \frac{1}{2}(q_1 + h(t, q_2, \dots, q_n))^2 + \frac{1}{2}q_2^2 + \dots + \frac{1}{2}q_n^2.$$

\* $\operatorname{ad}_{\mathbf{g}_i}^j \mathbf{g}_k$  oznacza iterowany nawias Liego pól wektorowych, zobacz dodatek A.3.

†Zobacz twierdzenie o prostowaniu, dodatek A.3.

4. Pola  $\mathbf{g}_k$ ,  $k = 1, \dots, m$  są takie, że dla funkcji  $W(\mathbf{q}) = \frac{1}{2} \sum_{i=2}^n q_i^2$  warunki

$$dW \mathbf{g}_k(\mathbf{q}) = \mathbf{0}, \quad k = 2, \dots, m,$$

oraz

$$\frac{\partial^j h(t, \mathbf{q})}{\partial t^j} = \mathbf{0}, \quad j \geq 1,$$

pociągają za sobą równość  $q_2 = q_3 = \dots = q_n = 0$ .

Wówczas, zachodzą następujące własności:

- a) Punkt  $\mathbf{0}$  jest globalnie jednostajnie asymptotycznie stabilnym punktem równowagi układu (11.46) ze sterowaniem

$$\begin{cases} u_1 = -\frac{\partial h(t, \mathbf{q})}{\partial t} - dV(t, \mathbf{q}) \mathbf{g}_1(\mathbf{q}) \\ u_2 = -dV(t, \mathbf{q}) \mathbf{g}_2(\mathbf{q}) \\ \vdots \\ u_m = -dV(t, \mathbf{q}) \mathbf{g}_m(\mathbf{q}), \end{cases} \quad (11.47)$$

gdzie  $dV(t, \mathbf{q}) = \left( \frac{\partial V(t, \mathbf{q})}{\partial q_1}, \dots, \frac{\partial V(t, \mathbf{q})}{\partial q_n} \right)$ .

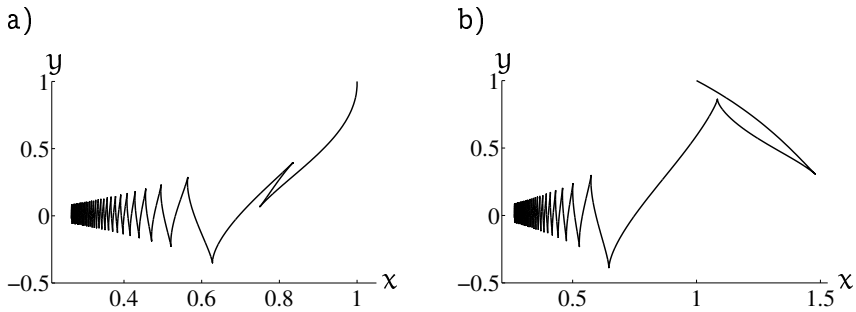
- b) Funkcja  $V(t, \mathbf{q})$  jest nierosnąca na trajektoriach nieautonomicznego układu dynamicznego (11.46), (11.47), tzn.

$$\dot{V}(t, \mathbf{q}(t)) \leq 0.$$

Przykładową funkcją  $h(t, \mathbf{q})$ , która może być zastosowana przy wyznaczaniu stabilizującego sprzężenia zwrotnego jest

$$h(t, \mathbf{q}) = \left( \sum_{i=2}^n q_i^2 \right) \cos t.$$

Przytoczone twierdzenie podaje sposób wyznaczenia algorytmu stabilizacji układu (11.47) z funkcją Lapunowa  $V(t, \mathbf{q})$ . Proponowane w nim stabilizujące sprzężenie zwrotne działa poprawnie przy spełnieniu przez układ sterowania (11.46) warunków 1 i 2 twierdzenia. W przypadku prostych kołowych robotów mobilnych wykazano, że dla modelu kinematyki dowolnego typu robota mobilnego można znaleźć taką zmianę układu oraz statyczne sprzężenie zwrotne, że powstały w ten sposób nowy układ sterowania spełnia założenia twierdzenia 11.3.7. Nietrudno zauważyć, że stabilizujące



**Rysunek 11.3** Trajektorie robota mobilnego Ulisses przy zastosowaniu algorytmu Corona-Pometa. Położenie początkowe wynosiło  $(x(0), y(0)) = (1, 1)$ , orientacja: a)  $\theta(0) = 0$ , b)  $\theta(0) = \pi/4$ .

sprężenie zwrotne postaci (11.47) jest gładką funkcją czasu, co w efekcie pozwala na konstrukcję algorytmu stabilizacji dla postaci normalnej (11.45) modelu dynamiki kołowego robota mobilnego. W tym celu, w myśl twierdzenia 11.3.7, należy najpierw wyznaczyć sterowanie  $\mathbf{u}(t)$ , a następnie obliczyć sterowanie  $\mathbf{v}(t)$  z zależności  $\mathbf{v} = \dot{\mathbf{u}}$ .

### Przykład 11.3.1 (Robot mobilny Ulisses)

Rozważmy zadanie stabilizacji stanu postaci normalnej modelu dynamiki robota mobilnego Ulisses

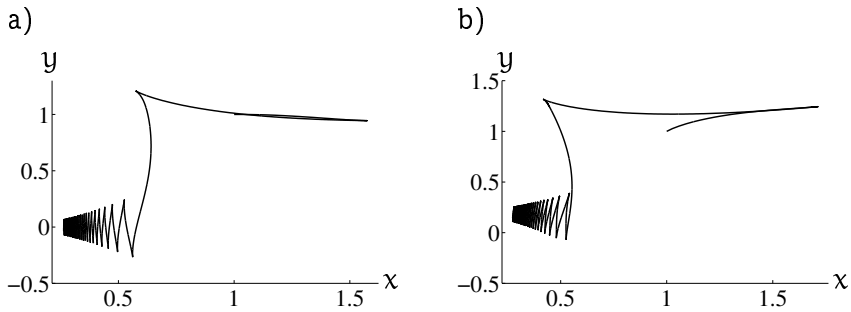
$$\begin{cases} \dot{x} = -\eta_1 \sin \theta \\ \dot{y} = \eta_1 \cos \theta \\ \dot{\theta} = \eta_2 \\ \dot{\eta}_1 = u_1 \\ \dot{\eta}_2 = u_2. \end{cases} \quad (11.48)$$

Zmienne  $\eta_1$  i  $\eta_2$  w równaniach (11.48) mają, odpowiednio, sens prędkości liniowej i prędkości zmian orientacji ruchu robota.

Rezultaty przykładowych symulacji przy sterowaniu według algorytmu Corona-Pometa przedstawiono na rysunkach 11.3 i 11.4<sup>‡</sup>. Zamieszczone symulacje pokazują, że istotną wadą algorytmu Corona-Pometa jest bardzo wolna zbieżność trajektorii do punktu docelowego. Ponadto generowane trajektorie charakteryzują się częstymi zmianami kierunku ruchu. ■

<sup>‡</sup>W przykładach prezentowanych w tym rozdziale symulacje wykonano w środowisku MATLAB<sup>®</sup> + SIMULINK<sup>®</sup>.





**Rysunek 11.4** Trajektorie robota mobilnego Ulisses przy zastosowaniu algorytmu Corona-Pometa. Położenie początkowe wynosiło  $(x(0), y(0)) = (1, 1)$ , orientacja: a)  $\theta(0) = \pi/2$ , b)  $\theta(0) = 3/4\pi$ .

### 11.3.2 Algorytm linearyzacji dynamicznej

W celu wprowadzenia podstawowych idei linearyzacji przy pomocy dynamicznego sprzężenia zwrotnego rozważmy afiniczny układ sterowania postaci

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u} = \mathbf{f}(\mathbf{x}) + \sum_{i=1}^m \mathbf{g}_i(\mathbf{x}_i)\mathbf{u}_i \quad (11.49)$$

i poddamy go następującym przekształceniom:

- dołączenie kompensatora dynamicznego

$$\begin{cases} \dot{\mathbf{z}} = \mathbf{K}(\mathbf{x}, \mathbf{z}) + \mathbf{L}(\mathbf{x}, \mathbf{z})\mathbf{v} \\ \mathbf{u} = \mathbf{M}(\mathbf{x}, \mathbf{z}) + \mathbf{N}(\mathbf{x}, \mathbf{z})\mathbf{v}, \end{cases} \quad (11.50)$$

gdzie  $\mathbf{z} \in \mathbb{R}^q$ ,  $\mathbf{v} \in \mathbb{R}^m$ ,

- statyczne sprzężenie zwrotne

$$\mathbf{v} = \boldsymbol{\gamma}(\mathbf{x}, \mathbf{z}) + \boldsymbol{\eta}(\mathbf{x}, \mathbf{z})\mathbf{w}, \quad (11.51)$$

- zmiana współrzędnych w rozszerzonej przestrzeni stanu  $(\mathbf{x}, \mathbf{z})$

$$\boldsymbol{\xi} = \boldsymbol{\varphi}(\mathbf{x}, \mathbf{z}). \quad (11.52)$$

Odwzorowania (11.50), (11.52) są gładkie, natomiast połączenie układu sterowania (11.49) i kompensatora (11.50), traktowane jako układ sterowania

z wejściem  $\mathbf{v}$  i wyjściem  $\mathbf{u}$ , powinno być odwracalne\*. Wprowadzenie współrzędnych stanu  $\xi$ , pozwala uzyskać nowy układ, określony w przestrzeni stanu o wymiarze  $n + q$ . Mówimy, że układ afiniczny (11.49) jest *dynamicznie linearyzowalny*, jeżeli istnieje kompensator dynamiczny (11.50), sprzężenie zwrotne (11.51) oraz układ współrzędnych  $\varphi$ , takie że po ich zastosowaniu układ (11.49) przyjmuje postać liniowego układu sterowania

$$\dot{\xi} = \mathbf{A}\xi + \mathbf{B}\mathbf{v}, \quad (11.53)$$

gdzie macierze  $\mathbf{A}$  i  $\mathbf{B}$  są odpowiednich rozmiarów. W przypadku, kiedy układ współrzędnych  $\varphi(\mathbf{x}, \mathbf{z})$  jest określony lokalnie, układ (11.49) nazywamy lokalnie dynamicznie linearyzowalnym.

### Przykład 11.3.2 (Robot mobilny Ulisses)

Przeanalizujmy zadanie śledzenia zadanej trajektorii położenia  $(x_d(t), y_d(t))$  dla postaci normalnej dynamiki robota Ulisses (11.48). Z uwagi na zadanie sterowania, które polega na śledzeniu trajektorii położenia, współrzędne  $(x, y)$  potraktujemy jako tzw. *wyjścia linearyzujące*. Przy takim wyborze wyjść linearyzujących, aby otrzymać dynamiczne sprzężenie zwrotne linearyzujące model (11.48), dwa pierwsze równania będziemy różniczkować tak długo, aż otrzymamy jawną zależność od sterowań

$$\begin{cases} \dot{x} = -\eta_1 \sin \theta \\ \ddot{x} = -u_1 \sin \theta - \eta_1 \eta_2 \cos \theta \\ \ddot{y} = -\dot{u}_1 \sin \theta - 2u_1 \eta_2 \cos \theta - u_2 \eta_1 \cos \theta + \eta_1 \eta_2^2 \sin \theta \\ \dot{y} = \eta_1 \cos \theta \\ \ddot{y} = u_1 \cos \theta - \eta_1 \eta_2 \sin \theta \\ \ddot{y} = \dot{u}_1 \cos \theta - 2u_1 \eta_2 \sin \theta - u_2 \eta_1 \sin \theta + \eta_1 \eta_2^2 \cos \theta. \end{cases} \quad (11.54)$$

Potraktujmy w powyższych wyrażeniach  $u_1$  jako nową zmienną stanu, wprowadźmy kompensator dynamiczny

$$\dot{u}_1 = v_1, \quad (11.55)$$

oraz podstawmy

$$u_2 = v_2. \quad (11.56)$$

\*Tzn. dla każdego stanu  $(\mathbf{x}, \mathbf{z})$  znajomość przebiegu wyjścia  $\mathbf{u}(\cdot)$  pozwala wyznaczyć sterowanie  $\mathbf{v}(\cdot)$  generujące to wyjście.

Wykonane przekształcenia pozwalają zapisać równania (11.54) w postaci

$$\begin{pmatrix} \ddot{x} \\ \ddot{y} \end{pmatrix} = \begin{bmatrix} -\sin \theta & -\eta_1 \cos \theta \\ \cos \theta & -\eta_1 \sin \theta \end{bmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} + \begin{pmatrix} -2u_1\eta_2 \cos \theta + \eta_1\eta_2^2 \sin \theta \\ -2u_1\eta_2 \sin \theta + \eta_1\eta_2^2 \cos \theta \end{pmatrix} = \\ = \mathbf{D}(\eta_1, \theta)\mathbf{v} + \boldsymbol{\alpha}(u_1, \eta_1, \eta_2, \theta).$$

Zauważmy, że macierz  $\mathbf{D}(\eta_1, \theta)$  stojąca przed nowymi zmiennymi sterującymi  $v_1$  i  $v_2$  jest nieosobliwa pod warunkiem, że  $\eta_1 \neq 0$ , co oznacza, że robot mobilny musi się poruszać z niezerową prędkością liniową. Przy spełnieniu warunku  $\eta_1 \neq 0$  możemy zastosować w układzie (11.54)–(11.56) statyczne sprzężenie zwrotne

$$\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = -\mathbf{D}^{-1}(\eta_1, \theta)\boldsymbol{\alpha}(u_1, \eta_1, \eta_2, \theta) + \mathbf{D}^{-1}(\eta_1, \theta)\mathbf{w}. \quad (11.57)$$

Ostatecznie dochodzimy do konkluzji, że układ (11.48) wraz z kompensatorem (11.55), (11.56) i sprzężeniem zwrotnym (11.57) ma we współrzędnych  $\boldsymbol{\xi} = \varphi(x, y, \theta, \eta_1, \eta_2, u_1)$  określonych zależnościami

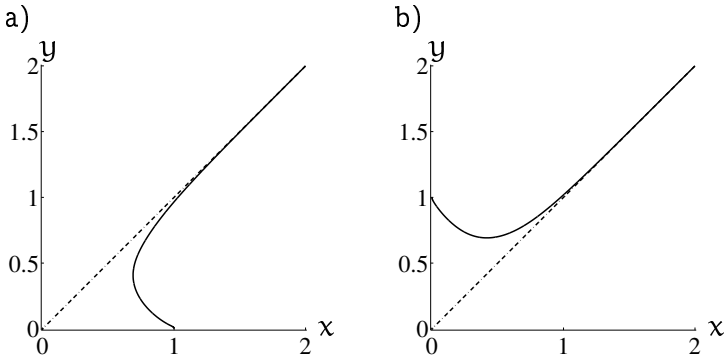
$$\begin{cases} \xi_1 = x \\ \xi_2 = -\eta_1 \sin \theta \\ \xi_3 = -u_1 \sin \theta - \eta_1\eta_2 \cos \theta \\ \xi_4 = y \\ \xi_5 = \eta_1 \cos \theta \\ \xi_6 = u_1 \cos \theta - \eta_1\eta_2 \sin \theta \end{cases}$$

postać liniową

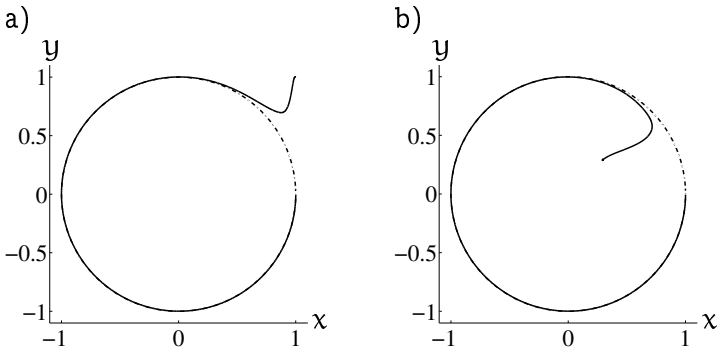
$$\begin{cases} \dot{\xi}_1 = \xi_2 & \dot{\xi}_4 = \xi_5 \\ \dot{\xi}_2 = \xi_3 & \dot{\xi}_5 = \xi_6 \\ \dot{\xi}_3 = w_1 & \dot{\xi}_6 = w_2. \end{cases} \quad (11.58)$$

Możemy zatem stwierdzić, że przy warunku  $\eta_1 \neq 0$  układ (11.48) jest dynamicznie linearyzowalny. Przy zastosowaniu klasycznych metod liniowej teorii regulacji łatwo otrzymać algorytm śledzenia trajektorii zadanej  $(x_d(t), y_d(t))$  w układzie (11.58):

$$\begin{cases} w_1 = \ddot{x}_d - k_2(\xi_3 - \ddot{x}_d) - k_1(\xi_2 - \dot{x}_d) - k_0(\xi_1 - x_d) \\ w_2 = \ddot{y}_d - l_2(\xi_6 - \ddot{y}_d) - l_1(\xi_5 - \dot{y}_d) - l_0(\xi_4 - y_d). \end{cases} \quad (11.59)$$



**Rysunek 11.5** Zachowanie robota mobilnego Ulisses przy zastosowaniu algorytmu linearyzacji dynamicznej do śledzenia zadanej trajektorii położenia (11.60).



**Rysunek 11.6** Zachowanie robota mobilnego Ulisses przy zastosowaniu algorytmu linearyzacji dynamicznej do śledzenia zadanej trajektorii położenia (11.61).

Na mocy kryterium Hurwitza, przy doborze wzmocnień spełniających warunki  $k_i, l_i, k_1 k_2, l_1 l_2 > 0$ ,  $i = 1, 2$ , błąd śledzenia zanika eksponencjalnie do zera. Algorytm linearyzacji dynamicznej kołowego robota mobilnego Ulisses zastosowaliśmy do śledzenia dwóch zadanych trajektorii położenia

$$\xi_d^1(t) = \begin{pmatrix} \xi_{1d}(t) \\ \xi_{4d}(t) \end{pmatrix} = \begin{pmatrix} x_d(t) \\ y_d(t) \end{pmatrix} = \begin{pmatrix} \frac{t}{5} \\ \frac{t}{5} \end{pmatrix}, \quad (\text{linia prosta}), \quad (11.60)$$

$$\xi_d^1(t) = \begin{pmatrix} \xi_{1d}(t) \\ \xi_{4d}(t) \end{pmatrix} = \begin{pmatrix} x_d(t) \\ y_d(t) \end{pmatrix} = \begin{pmatrix} r \cos \frac{t}{5} \\ r \sin \frac{t}{5} \end{pmatrix}, \quad r = 1[\text{m}], \quad (\text{okrąg}), \quad (11.61)$$

przy różnych warunkach początkowych. Wybrane wyniki symulacji komputerowych zostały przedstawione na rysunkach 11.5 i 11.6. Jak widać,

algorytm sterowania oparty na metodzie linearyzacji dynamicznej znakomicie nadaje się do śledzenia trajektorii zadanej wyjść linearyzujących. Niewątpliwą jego zaletą jest globalna eksponencjalna stabilność. Należy jednak pamiętać, że śledzenie w przypadku robota mobilnego Ulisses dotyczy jedynie trajektorii wyjść linearyzujących oraz że algorytm wymaga, aby robot był w ruchu. ■

### 11.3.3 Algorytm Walsh-Tilbury’ego-Sastry’ego-Murray-Laumonda

Rozważmy postać normalną dynamiki kołowego robota mobilnego (11.45), którą przy oznaczeniach  $\mathbf{x} = \begin{pmatrix} \mathbf{q} \\ \boldsymbol{\eta} \end{pmatrix}$ ,  $\mathbf{f}(\mathbf{x}) = \begin{pmatrix} \mathbf{G}(\mathbf{q})\boldsymbol{\eta} \\ \mathbf{0} \end{pmatrix}$ ,  $\mathbf{g}(\mathbf{x}) = \begin{bmatrix} \mathbf{0} \\ \mathbb{I}_m \end{bmatrix}$  można przedstawić w postaci afinicznego układu sterowania

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}, \quad \mathbf{x} \in \mathbb{R}^n. \quad (11.62)$$

Założmy, że  $\mathbf{x}_d(t)$  oznacza trajektorię zadaną, natomiast  $\mathbf{u}_d(t)$  — sterowanie odpowiadające tej trajektorii. Przybliżenie liniowe modelu (11.62) wzdłuż trajektorii zadanej jest nieautonomicznym liniowym układem sterowania postaci

$$\dot{\boldsymbol{\xi}} = \mathbf{A}(t)\boldsymbol{\xi} + \mathbf{B}(t)\mathbf{v}, \quad (11.63)$$

gdzie

$$\mathbf{A}(t) = \frac{\partial \mathbf{f}(\mathbf{x}_d(t))}{\partial \mathbf{x}} + \frac{\partial (\mathbf{g}(\mathbf{x}_d(t))\mathbf{u}_d(t))}{\partial \mathbf{x}}, \quad \mathbf{B}(t) = \mathbf{g}(\mathbf{x}_d(t)).$$

Niech macierz  $\boldsymbol{\Phi}(t, s)$  będzie rozwiązaniem równania różniczkowego

$$\frac{d}{dt}\boldsymbol{\Phi}(t, s) = \mathbf{A}(t)\boldsymbol{\Phi}(t, s)$$

przy warunku początkowym  $\boldsymbol{\Phi}(s, s) = \mathbb{I}_n$ .

Dla pewnego  $\alpha > 0$  zdefiniujemy macierz\*

$$\mathbf{H}_c(s, t) = \int_s^t e^{6\alpha(s-\tau)} \boldsymbol{\Phi}(s, \tau) \mathbf{B}(\tau) \mathbf{B}^\top(\tau) \boldsymbol{\Phi}^\top(s, \tau) d\tau.$$

Jeżeli istnieje takie  $\delta > 0$ , że macierz  $\mathbf{H}_c(t, t + \delta)$  jest nieosobliwa dla każdej chwili  $t$ , wówczas macierz

$$\mathbf{P}_c(t) = \mathbf{H}_c^{-1}(t, t + \delta)$$

jest dobrze określona.

\*Porównaj z macierzą Grama (4.4).

Przyjęte założenia pozwalają udowodnić następujące twierdzenie.

**Twierdzenie 11.3.8** *Jeżeli istnieją liczby  $p_c^m, p_c^M > 0$ , takie że*

$$\forall t \in \mathbb{R}_+ \quad 0 < p_c^m \mathbf{z}^T \mathbf{z} < \mathbf{z}^T \mathbf{P}_c(t) \mathbf{z} < p_c^M \mathbf{z}^T \mathbf{z},$$

*wtedy, dla dowolnej ciągłej i ograniczonej funkcji  $\gamma : \mathbb{R}_+ \rightarrow [\frac{1}{2}, \infty)$ , istnieje zależne od czasu sprzężenie zwrotne postaci*

$$\mathbf{u} = \mathbf{u}_d(t) - \gamma(t) \mathbf{B}^T(t) \mathbf{P}_c(t) (\mathbf{x} - \mathbf{x}_d), \quad (11.64)$$

*które zapewnia lokalne śledzenie trajektorii  $\mathbf{x}_d(t)$  w układzie sterowania (11.62), z błędem zbieżnym do zera jednostajnie i wykładniczo, przy czym prędkość zbieżności jest większa od*

$$2\alpha p_c^m (p_c^M)^{-1} > 0.$$

Ze względu na to, że algorytm (11.64) wymaga znajomości trajektorii zadanej w przyszłości, używa się modyfikacji tego algorytmu, która polega na wprowadzeniu macierzy

$$\mathbf{P}_r(t) = \mathbf{H}_c^{-1}(t, t - \delta),$$

która zależy wyłącznie od wartości trajektorii zadanej w przeszłości. Jeżeli jest znana macierz  $\mathbf{P}_r(t)$ , algorytm sterowania definiujemy w następujący sposób.

**Twierdzenie 11.3.9** *Jeżeli istnieją liczby  $p_r^m, p_r^M > 0$  spełniające nierówność*

$$\forall t \in \mathbb{R}_+ \quad 0 < p_r^m \mathbf{z}^T \mathbf{z} < \mathbf{z}^T \mathbf{P}_r(t) \mathbf{z} < p_r^M \mathbf{z}^T \mathbf{z},$$

*wtedy, dla dowolnej ciągłej i ograniczonej funkcji  $\gamma : \mathbb{R}_+ \rightarrow [\frac{1}{2}, \infty)$ , istnieje zależne od czasu sprzężenie zwrotne postaci*

$$\mathbf{u} = \mathbf{u}_d(t) - \gamma(t) \mathbf{B}^T(t) \mathbf{P}_r(t) (\mathbf{x} - \mathbf{x}_d), \quad (11.65)$$

*które zapewnia lokalną jednostajną eksponencjalną stabilność śledzenia trajektorii zadanej  $\mathbf{x}_d(t)$  w układzie (11.62), z prędkością zbieżności błędu śledzenia większą od  $\alpha$ .*

Algorytm śledzenia Walsha-Tilbury'ego-Sastry'ego-Murraya-Laumonda jest złożony obliczeniowo. Nawet w tak prostym przypadku jak model dynamiki robota Ulisses przy trajektorii zadanej będącej łukiem okręgu, elementy

macierzy  $\mathbf{H}_c$  mają na tyle skomplikowaną postać, że obliczenie w postaci symbolicznej elementów macierzy  $\mathbf{P}_r$  i wektora sterowania  $\mathbf{u}(t)$  jest praktycznie niemożliwe. Wadą algorytmu jest także jego lokalna zbieżność<sup>†</sup>. Do zalet algorytmu Walsha-Tilbury'ego-Sastry'ego-Murraya-Laumonda należy możliwość śledzenia zarówno współrzędnych położenia, jak i orientacji robota.

### 11.3.4 Sterowanie we współrzędnych linearyzujących

Rozważmy ograniczenia fazowe w postaci Pfaffa

$$\mathbf{A}(\mathbf{q})\dot{\mathbf{q}} = \mathbf{0} \quad (11.66)$$

określające kinematykę kołowego robota mobilnego (11.8)

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u} = \sum_{i=1}^m \mathbf{g}_i(\mathbf{q})u_i. \quad (11.67)$$

Wybermy odwzorowanie  $\mathbf{h} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ , którego macierz Jacobiego  $\frac{\partial \mathbf{h}}{\partial \mathbf{q}}(\mathbf{q})$  ma rząd  $m$  i uzupełnijmy je odwzorowaniem  $\mathbf{k} : \mathbb{R}^n \rightarrow \mathbb{R}^l$  spełniającym warunek

$$\text{rank} \begin{bmatrix} \frac{\partial \mathbf{h}}{\partial \mathbf{q}} \\ \frac{\partial \mathbf{k}}{\partial \mathbf{q}} \end{bmatrix}(\mathbf{q}) = n.$$

Przy takim wyborze przekształcenie

$$\boldsymbol{\xi} = \boldsymbol{\varphi}(\mathbf{q}) = \begin{pmatrix} \boldsymbol{\xi}^1 \\ \boldsymbol{\xi}^2 \end{pmatrix} = \begin{pmatrix} \mathbf{h}(\mathbf{q}) \\ \mathbf{k}(\mathbf{q}) \end{pmatrix} \quad (11.68)$$

jest lokalnym dyfeomorfizmem przestrzeni stanu. Z powodów, które staną się jasne nieco później, nowe współrzędne  $\boldsymbol{\xi}$  będziemy nazywać *linearyzującymi*. We współrzędnych linearyzujących ograniczenia fazowe (11.66) przyjmą postać

$$\bar{\mathbf{A}}(\boldsymbol{\xi})\dot{\boldsymbol{\xi}} = \mathbf{0}. \quad (11.69)$$

W celu wyznaczenia macierzy  $\bar{\mathbf{A}}(\boldsymbol{\xi})$  zastosujemy następujące rozumowanie. Biorąc pod uwagę zależność (11.68), formułę (11.69) możemy przepisać jako

$$\bar{\mathbf{A}}(\boldsymbol{\varphi}(\mathbf{q})) \begin{bmatrix} \frac{\partial \mathbf{h}}{\partial \mathbf{q}} \\ \frac{\partial \mathbf{k}}{\partial \mathbf{q}} \end{bmatrix} \dot{\mathbf{q}} = \mathbf{0}.$$

<sup>†</sup> Symulacje pokazują, że przy dużym błędzie początkowym algorytm nie zapewnia zbieżności do trajektorii zadanej.

Wobec zależności (11.66), musi istnieć macierz  $\mathbf{M}(\mathbf{q})$ , taka że

$$\bar{\mathbf{A}}(\boldsymbol{\varphi}(\mathbf{q})) \begin{bmatrix} \frac{\partial \mathbf{h}}{\partial \mathbf{q}} \\ \frac{\partial \mathbf{k}}{\partial \mathbf{q}} \end{bmatrix} = \mathbf{M}(\mathbf{q})\mathbf{A}(\mathbf{q}). \quad (11.70)$$

Ponieważ  $\mathbf{A}(\mathbf{q})\mathbf{G}(\mathbf{q}) = \mathbf{0}$ , po pomnożeniu obu stron równości (11.70) prawostronnie przez  $\mathbf{G}(\mathbf{q})$  otrzymujemy

$$\bar{\mathbf{A}}(\boldsymbol{\varphi}(\mathbf{q})) \begin{bmatrix} \frac{\partial \mathbf{h}}{\partial \mathbf{q}} \mathbf{G}(\mathbf{q}) \\ \frac{\partial \mathbf{k}}{\partial \mathbf{q}} \mathbf{G}(\mathbf{q}) \end{bmatrix} = \mathbf{0}. \quad (11.71)$$

Zauważmy, że macierz  $\frac{\partial \mathbf{h}}{\partial \mathbf{q}} \mathbf{G}(\mathbf{q})$  jest nieosobliwa. Podzielimy teraz macierz  $\bar{\mathbf{A}}$  na bloki  $\bar{\mathbf{A}}_1$  i  $\bar{\mathbf{A}}_2$  rozmiarów  $l \times m$  oraz  $l \times l$ . Lokalnie\*  $\bar{\mathbf{A}}_2$  jest rzędu  $l$ . Zależność (11.71) pozwala wyliczyć macierz  $\bar{\mathbf{A}}_1$  jako

$$\bar{\mathbf{A}}_1(\boldsymbol{\varphi}(\mathbf{q})) = -\bar{\mathbf{A}}_2(\boldsymbol{\varphi}(\mathbf{q})) \frac{\partial \mathbf{k}}{\partial \mathbf{q}} \mathbf{G}(\mathbf{q}) \left( \frac{\partial \mathbf{h}}{\partial \mathbf{q}} \mathbf{G}(\mathbf{q}) \right)^{-1},$$

skąd wynika, że ograniczenia fazowe (11.69) można przedstawić w postaci

$$\begin{bmatrix} -\mathbf{H}(\boldsymbol{\xi}) & \mathbb{I}_l \end{bmatrix} \dot{\boldsymbol{\xi}} = \mathbf{0}, \quad (11.72)$$

z macierzą  $\mathbf{H}(\boldsymbol{\xi}) = \frac{\partial \mathbf{k}}{\partial \mathbf{q}} \mathbf{G}(\boldsymbol{\varphi}^{-1}(\boldsymbol{\xi})) \left( \frac{\partial \mathbf{h}}{\partial \mathbf{q}} \mathbf{G}(\boldsymbol{\varphi}^{-1}(\boldsymbol{\xi})) \right)^{-1}$ , zaś kinematykę (11.67) jako

$$\dot{\boldsymbol{\xi}} = \mathbf{L}(\boldsymbol{\xi})\mathbf{v}, \quad (11.73)$$

gdzie  $\mathbf{L}(\boldsymbol{\xi}) = \begin{bmatrix} \mathbb{I}_m \\ \mathbf{H}(\boldsymbol{\xi}) \end{bmatrix}$ ,  $\mathbf{v} \in \mathbb{R}^m$ . Zmiana układu współrzędnych  $\boldsymbol{\xi} = \boldsymbol{\varphi}(\mathbf{q})$  zależy jedynie od postaci ograniczeń fazowych (własności geometrycznych), zatem współrzędne te mogą być używane pomimo nieznanności parametrów dynamiki. Równania dynamiki robota mobilnego we współrzędnych  $\boldsymbol{\xi}$ , podlegającego więzom (11.72) zapiszemy zgodnie z Zasadą d'Alemberta

$$\bar{\mathbf{Q}}(\boldsymbol{\xi})\ddot{\boldsymbol{\xi}} + \bar{\mathbf{C}}(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}})\dot{\boldsymbol{\xi}} + \bar{\mathbf{D}}(\boldsymbol{\xi}) = \bar{\mathbf{B}}(\boldsymbol{\xi})\mathbf{u} + \begin{bmatrix} -\mathbf{H}^T(\boldsymbol{\xi}) \\ \mathbb{I}_l \end{bmatrix} \boldsymbol{\lambda}, \quad (11.74)$$

$\boldsymbol{\lambda} \in \mathbb{R}^l$ . Standardowa procedura eliminacji wektora mnożników Lagrange'a

\*Przy analitycznej zależności  $\bar{\mathbf{A}}(\mathbf{q})$  — prawie wszędzie.



oraz wykorzystanie równości

$$\ddot{\xi} = \begin{bmatrix} \mathbb{I}_m \\ \mathbf{H}(\xi) \end{bmatrix} \dot{\mathbf{v}} + \begin{bmatrix} \mathbf{0} \\ \dot{\mathbf{H}}(\xi) \end{bmatrix} \mathbf{v},$$

proceedzi do równań dynamiki postaci

$$\mathbf{J}(\xi)\dot{\mathbf{v}} + \mathbf{C}(\xi, \mathbf{v})\mathbf{v} + \mathbf{D}(\xi) = \mathbf{B}(\xi)\mathbf{u}, \quad (11.75)$$

w których  $\mathbf{J}(\xi) = \mathbf{L}^\top(\xi)\bar{\mathbf{Q}}(\xi)\mathbf{L}(\xi)$ ,  $\mathbf{B}(\xi) = \mathbf{L}^\top(\xi)\bar{\mathbf{B}}(\xi)$ ,  $\mathbf{D}(\xi) = \mathbf{L}^\top(\xi)\bar{\mathbf{D}}(\xi)$ ,  $\mathbf{C}(\xi, \mathbf{v}) = \mathbf{L}^\top(\xi)\left(\bar{\mathbf{C}}(\xi, \mathbf{L}(\xi)\mathbf{v})\mathbf{L}(\xi) + \bar{\mathbf{Q}}(\xi)\dot{\mathbf{L}}(\xi)\right)$ . Ostatecznie, z połączenia wyrażen (11.73) i (11.75), otrzymujemy równania kinematyki i dynamiki robota mobilnego we współrzędnych linearyzujących  $\xi$

$$\begin{cases} \dot{\xi}^1 = \mathbf{v} & (11.76) \\ \mathbf{J}(\xi)\dot{\mathbf{v}} + \mathbf{C}(\xi, \mathbf{v})\mathbf{v} + \mathbf{D}(\xi) = \mathbf{B}(\xi)\mathbf{u} & (11.77) \\ \dot{\xi}^2 = \mathbf{H}(\xi)\mathbf{v}. & (11.78) \end{cases}$$

Nietrudno zauważyć, że macierze  $\mathbf{H}(\xi)$  i  $\mathbf{B}(\xi)$  są zależne jedynie od parametrów geometrycznych robota mobilnego. Zależność od parametrów dynamiki macierzy inercji  $\mathbf{J}(\xi)$ , macierzy  $\mathbf{C}(\xi, \mathbf{v})$  i wektora  $\mathbf{D}(\xi)$  może być przedstawiona w następującej postaci:

$$\begin{aligned} \mathbf{J}(\xi) &= \mathbf{J}_0(\xi) + \sum_{i=1}^p \mathbf{J}_i(\xi)\theta_i, \\ \mathbf{C}(\xi, \mathbf{v}) &= \mathbf{C}_0(\xi, \mathbf{v}) + \sum_{i=1}^p \mathbf{C}_i(\xi, \mathbf{v})\theta_i, \\ \mathbf{D}(\xi) &= \mathbf{D}_0(\xi) + \sum_{i=1}^p \mathbf{D}_i(\xi)\theta_i. \end{aligned}$$

Parametry  $\theta_i$ ,  $i = 1, 2, \dots, p$ , występujące w powyższych zależnościach nazywamy *barycentrycznymi*.

Nietrudno wykazać, że macierz  $\dot{\mathbf{J}}(\xi) - 2\mathbf{C}(\xi, \mathbf{v})$  jest skośnie symetryczna na trajektoriach układu (11.76)–(11.78). Macierze  $\mathbf{H}(\xi)$ ,  $\mathbf{J}(\xi)$ ,  $\mathbf{D}(\xi)$  i  $\mathbf{B}(\xi)$  są ograniczone (klasy  $\mathbb{B}$ ), a macierz  $\mathbf{C}(\xi, \mathbf{v})$  spełnia warunek  $\|\mathbf{C}(\xi, \mathbf{v})\| \leq C\|\mathbf{v}\|$ . Wynika stąd, że model dynamiki kołowego robota mobilnego we współrzędnych linearyzujących (11.76)–(11.77) ma podobne własności strukturalne do modelu dynamiki manipulatora (układu holonomicznego) o  $m$  stopniach

swobody<sup>†</sup>. Różnica polega jednak na tym, że model dynamiki robota mobilnego zawiera dodatkowo podsystem opisany równaniem (11.78).

Zostało udowodnione, że dla każdego z pięciu podstawowych typów robotów mobilnych (zobacz twierdzenie 11.2.1) można skonstruować wyjścia linearyzujące w taki sposób, aby składowe wektora  $\xi^1$  były nowymi współrzędnymi położenia lokalnego układu odniesienia umieszczonego na nadwoziu robota, natomiast składowe wektora  $\xi^2$  opisywały orientację robota, kąty obrotu kół oraz kąty orientacji kół w lokalnym układzie odniesienia.

### Przykład 11.3.3 (Robot mobilny Ulisses)

Dla robota mobilnego Ulisses proponujemy następującą definicję współrzędnych linearyzujących  $\xi^1$  i  $\xi^2$

$$\xi^1 = \begin{pmatrix} x - e \sin \theta \\ y + e \cos \theta \end{pmatrix} = \begin{pmatrix} \xi_{11} \\ \xi_{12} \end{pmatrix}, \quad \xi^2 = \begin{pmatrix} \theta \\ \phi_1 \\ \phi_2 \end{pmatrix}, \quad e \neq 0, \quad (11.79)$$

i obliczamy macierz

$$\mathbf{H}(\xi) = \begin{bmatrix} -\frac{1}{e} \cos \theta & -\frac{1}{e} \sin \theta \\ \frac{1}{r} \sin \theta + \frac{1}{er} \cos \theta & -\frac{1}{r} \cos \theta + \frac{1}{er} \sin \theta \\ -\frac{1}{r} \sin \theta + \frac{1}{er} \cos \theta & \frac{1}{r} \cos \theta + \frac{1}{er} \sin \theta \end{bmatrix}.$$

Jak łatwo zauważyć, macierz  $\mathbf{H}(\xi)$  jest ograniczona. Składowe wektora  $\xi^1$  opisują nowe położenie lokalnego układu odniesienia, natomiast składowe wektora  $\xi^2$  — orientację robota i kąty obrotu kół. Energia kinetyczna robota mobilnego we współrzędnych  $\xi$

$$E_k(\xi, \dot{\xi}) = \frac{1}{2}(I_z + 2m_k l^2)\dot{\theta}^2 + \frac{1}{2}(m + 2m_k) \left( \dot{\xi}_{11}^2 + 2\dot{\xi}_{11}\dot{\theta}e \cos \theta + \dot{\xi}_{12}^2 + 2\dot{\xi}_{12}\dot{\theta}e \sin \theta + e^2\dot{\theta}^2 \right) + \frac{1}{2}I_k (\dot{\phi}_1^2 + \dot{\phi}_2^2),$$

gdzie

$m$  — masa nadwozia robota,

$I_z$  — moment bezwładności nadwozia robota względem osi  $Z$  układu  $X_1 Y_1 Z_1$ ,

$m_k$  — masa koła,

<sup>†</sup>Zobacz podrozdział 5.2.

$I_k$  — moment bezwładności koła względem osi  $\times$  odpowiedniego układu lokalnego,

$m + 2m_k$  — masa całkowita robota,

$I_z + 2m_k l^2$  — całkowity moment bezwładności robota.

Przy wyborze parametrów barycentrycznych  $\theta_1 = I_k$ ,  $\theta_2 = m + 2m_k$ ,  $\theta_3 = I_z + 2m_k l^2$ , macierze  $\mathbf{J}(\boldsymbol{\xi})$ ,  $\mathbf{C}(\boldsymbol{\xi}, \mathbf{v})$  i  $\mathbf{B}(\boldsymbol{\xi})$  można przedstawić jako:

$$\mathbf{J}(\boldsymbol{\xi}) = \sum_{i=1}^3 \mathbf{J}_i(\boldsymbol{\xi})\theta_i,$$

$$\mathbf{C}(\boldsymbol{\xi}, \mathbf{v}) = \sum_{i=1}^3 \mathbf{C}_i(\boldsymbol{\xi}, \mathbf{v})\theta_i,$$

$$\mathbf{B}(\boldsymbol{\xi}) = \begin{bmatrix} \frac{1}{r} \sin \theta + \frac{1}{er} \cos \theta & -\frac{1}{r} \sin \theta + \frac{1}{er} \cos \theta \\ -\frac{1}{r} \cos \theta + \frac{1}{er} \sin \theta & \frac{1}{r} \cos \theta + \frac{1}{er} \sin \theta \end{bmatrix},$$

przy oznaczeniach

$$\mathbf{J}_1(\boldsymbol{\xi}) = \begin{bmatrix} \left( \sin^2 \theta + \frac{l^2}{e^2} \cos^2 \theta \right) \frac{2}{r^2} & \sin \theta \cos \theta \left( \frac{l^2}{e^2} - 1 \right) \frac{2}{r^2} \\ \sin \theta \cos \theta \left( \frac{l^2}{e^2} - 1 \right) \frac{2}{r^2} & \left( \cos^2 \theta + \frac{l^2}{e^2} \sin^2 \theta \right) \frac{2}{r^2} \end{bmatrix},$$

$$\mathbf{J}_2(\boldsymbol{\xi}) = \begin{bmatrix} \sin^2 \theta & -\sin \theta \cos \theta \\ -\sin \theta \cos \theta & \cos^2 \theta \end{bmatrix}, \quad \mathbf{J}_3(\boldsymbol{\xi}) = \begin{bmatrix} \frac{1}{e^2} \cos^2 \theta & \frac{1}{e^2} \sin \theta \cos \theta \\ \frac{1}{e^2} \sin \theta \cos \theta & \frac{1}{e^2} \sin^2 \theta \end{bmatrix},$$

$$\mathbf{C}_1(\boldsymbol{\xi}, \mathbf{v}) = \begin{bmatrix} -\frac{2}{er^2} \sin \theta \cos \theta \left( 1 - \frac{l^2}{e^2} \right) & -\frac{2}{er^2} \left( \sin^2 \theta + \frac{l^2}{e^2} \cos^2 \theta \right) \\ (v_1 \cos \theta + v_2 \sin \theta) & (v_1 \cos \theta + v_2 \sin \theta) \\ \frac{2}{er^2} \left( \cos^2 \theta + \frac{l^2}{e^2} \sin^2 \theta \right) & -\frac{2}{er^2} \sin \theta \cos \theta \left( -1 + \frac{l^2}{e^2} \right) \\ (v_1 \cos \theta + v_2 \sin \theta) & (v_1 \cos \theta + v_2 \sin \theta) \end{bmatrix},$$

$$\mathbf{C}_2(\boldsymbol{\xi}, \mathbf{v}) = \begin{bmatrix} -\frac{1}{e} \sin \theta \cos \theta & -\frac{1}{2e} (\sin^2 \theta - \cos^2 \theta) \\ (v_1 \cos \theta + v_2 \sin \theta) & (v_1 \cos \theta + v_2 \sin \theta) \\ -\frac{1}{2e} (\sin^2 \theta - \cos^2 \theta) & \frac{1}{e} \sin \theta \cos \theta \\ (v_1 \cos \theta + v_2 \sin \theta) & (v_1 \cos \theta + v_2 \sin \theta) \end{bmatrix},$$

$$\mathbf{C}_3(\boldsymbol{\xi}, \mathbf{v}) = \begin{bmatrix} \frac{1}{e^3} \sin \theta \cos \theta & -\frac{1}{e^3} \cos^2 \theta \\ (v_1 \cos \theta + v_2 \sin \theta) & (v_1 \cos \theta + v_2 \sin \theta) \\ \frac{1}{e^3} \sin^2 \theta & -\frac{1}{e^3} \sin \theta \cos \theta \\ (v_1 \cos \theta + v_2 \sin \theta) & (v_1 \cos \theta + v_2 \sin \theta) \end{bmatrix}.$$

Łatwo sprawdzić, że w modelu dynamiki kołowego robota mobilnego *Ulyses* macierze  $\mathbf{J}(\boldsymbol{\xi})$ ,  $\mathbf{C}(\boldsymbol{\xi}, \mathbf{v})$  i  $\mathbf{B}(\boldsymbol{\xi})$  są ograniczone ze względu na  $\boldsymbol{\xi}$ , a norma macierzy  $\mathbf{C}(\boldsymbol{\xi}, \mathbf{v})$  jest ograniczona przez liniową funkcję normy  $\|\mathbf{v}\|$ . ■

Celem naszych dalszych rozważań będzie następujące zadanie sterowania układu (11.76)–(11.78):

Mając daną trajektorię  $\xi_d^1(t)$ , klasy  $C^2$ , znaleźć sterowanie  $\mathbf{u}(t)$ , takie że w układzie (11.76)–(11.78) błąd śledzenia trajektorii położenia  $\mathbf{e}(t) = \xi^1(t) - \xi_d^1(t)$  dąży do  $\mathbf{0}$  przy  $t \rightarrow +\infty$ , a prędkość  $\dot{\xi}^2(t)$  jest ograniczona (wszędzie lub prawie wszędzie).

Założmy, że składowe wektora  $\xi^1(t)$  opisują położenie lokalnego układu odniesienia związanego z poruszającym się robotem, natomiast wektor  $\xi^2(t)$  składa się ze współrzędnych opisujących orientację robota, kąty obrotu kół oraz orientacje kół względem lokalnego układu odniesienia. Tym sposobem rozpatrywane zadanie sterowania sprowadza się do wymagania śledzenia położenia robota we współrzędnych  $\xi^1(t)$ , natomiast ograniczoność wektora  $\dot{\xi}^2(t)$  zapewni ograniczoność prędkości zmian orientacji robota i prędkości obrotu kół. Zauważmy, że jeżeli macierz sterowań  $\bar{\mathbf{B}}(\xi)$  w wyrażeniu (11.74) jest pełnego rzędu, to macierz  $\mathbf{B}(\xi)$  w układzie (11.76)–(11.78) jest nieosobliwa. W takiej sytuacji i przy założeniu, że parametry barycentryczne układu (11.77) są znane, zastosowanie sprzężenia zwrotnego

$$\mathbf{u}(\xi, \mathbf{v}, \mathbf{w}) = \mathbf{B}^{-1}(\xi) (\mathbf{J}(\xi)\mathbf{w} + \mathbf{C}(\xi, \mathbf{v})\mathbf{v} + \mathbf{D}(\xi)), \quad (11.80)$$

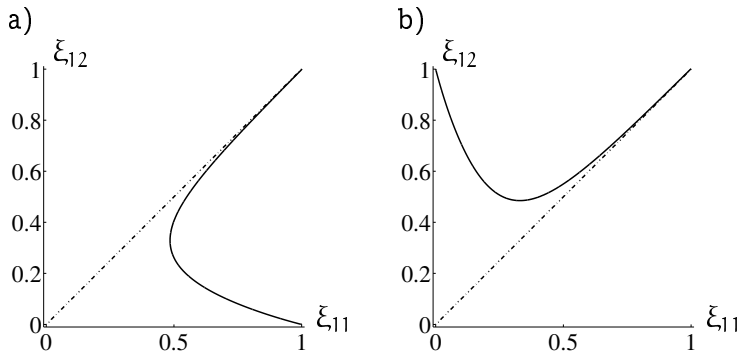
w którym  $\mathbf{w} \in \mathbb{R}^m$  oznacza nowy wektor sterowań, pozwala na linearyzację podsystemu (11.76)–(11.77)

$$\begin{cases} \dot{\xi}^1 = \mathbf{v} \\ \dot{\mathbf{v}} = \mathbf{w} \\ \dot{\xi}^2 = \mathbf{H}(\xi)\mathbf{v}. \end{cases} \quad (11.81)$$

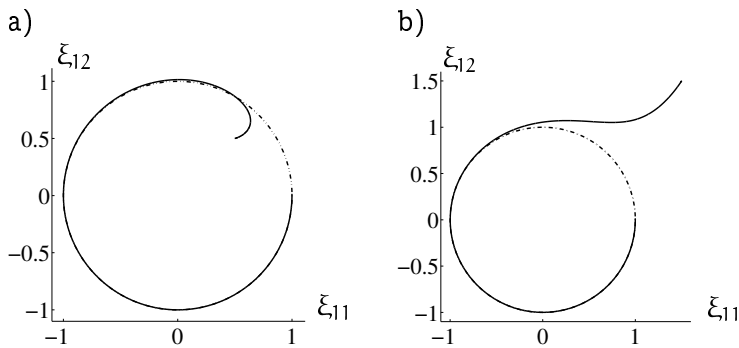
Własność ta tłumaczy nazwę *współrzędne linearyzujące*. Jeżeli  $\xi_d^1(\cdot) \in C^2$ , to wybierając sterowanie  $\mathbf{w}$  w postaci regulatora PD z korekcją

$$\mathbf{w} = \ddot{\xi}_d^1 - \mathbf{R}_1 \dot{\mathbf{e}} - \mathbf{R}_0 \mathbf{e}, \quad (11.82)$$

gdzie  $\mathbf{e} = \xi^1 - \xi_d^1$ , z diagonalnymi, dodatnio określonymi macierzami wzmocnień  $\mathbf{R}_0$ ,  $\mathbf{R}_1$ , uzyskujemy eksponencjalną stabilność błędu śledzenia  $\mathbf{e}$  oraz jego pochodnej  $\dot{\mathbf{e}}$ . Dalsza analiza zachowania układu (11.81) ze sterowaniem (11.82) pozwala wykazać następujący rezultat.



**Rysunek 11.7** Zachowanie robota mobilnego Ulisses przy śledzeniu zadanej trajektorii położenia (11.60) we współrzędnych liniaryzujących.



**Rysunek 11.8** Zachowanie robota mobilnego Ulisses przy śledzeniu zadanej trajektorii położenia (11.61) we współrzędnych liniaryzujących.

**Twierdzenie 11.3.10** Załóżmy, że prędkości zadane  $\dot{\xi}_d^1(\cdot) \in \mathbb{B}$  i że macierz  $\mathbf{H}(\xi)$  w układzie (11.81) jest ograniczona. Wówczas, prędkości  $\dot{\xi}^2(t)$  w układzie (11.81) ze sterowaniem (11.82) są ograniczone,  $\dot{\xi}^2(\cdot) \in \mathbb{B}$ , oraz istnieją liczby  $a, b$ , takie że współrzędne  $\|\dot{\xi}^2(t)\| \leq a + bt$ .

#### Przykład 11.3.4 (Robot mobilny Ulisses)

Algorytm sterowania przy użyciu wyjść liniaryzujących zastosowaliśmy do robota Ulisses. Przykładowe wyniki badań symulacyjnych przedstawiliśmy na rysunkach 11.7 i 11.8. Macierze wzmocnień wybraliśmy równe  $\mathbf{R}_0 = \mathbf{R}_1 = \text{diag}\{5\}$ . Przedmiotem badań było zachowanie robota mobilnego przy różnych zadanych trajektoriach położenia i stanach początkowych. Symulacje potwierdziły eksponencjalną zbieżność błędu śledzenia  $\mathbf{e}$  oraz ograniczoność orientacji  $\theta(t)$  robota mobilnego. ■

### 11.3.5 Sterowanie adaptacyjne we współrzędnych linearyzujących

Algorytmy sterowania kołowych robotów mobilnych opisane w poprzednich podrozdziałach wykorzystywały pełną znajomość modeli kinematyki i dynamiki. Obecnie założymy, że parametry barycentryczne modelu dynamiki (11.76)–(11.78) są nieznanne i rozważymy następujące zadanie sterowania adaptacyjnego.

Znaleźć prawo sterowania i algorytm estymacji parametrów dynamiki, które w układzie opisanym równaniami (11.76)–(11.78) zapewniają śledzenie trajektorii zadanej  $\xi_d^1(t)$  i ograniczoność prędkości  $\dot{\xi}^2(t)$ .

Z uwagi na podobieństwo strukturalne układu sterowania (11.76)–(11.77) do modelu dynamiki robota manipulacyjnego możemy przypuszczać, że niektóre spośród algorytmów sterowania adaptacyjnego robotów manipulacyjnych przedstawionych w rozdziale 6.2 dostarczą poprawnego rozwiązania adaptacyjnego zadania sterowania kołowych robotów mobilnych. Okazuje się, że tak jest istotnie, prawdziwy bowiem jest następujący rezultat.

**Twierdzenie 11.3.11** *Niech trajektoria zadana  $\xi_d^1(\cdot) \in \mathbb{C}^2$  i niech macierz  $H(\xi)$  w układzie (11.76)–(11.77) będzie ograniczona. Wówczas, jeżeli dla każdego  $\xi^2$  algorytm sterowania adaptacyjnego zastosowany do układu (11.76)–(11.77) zapewnia ograniczoność i asymptotyczną stabilność błędu śledzenia  $e = \xi^1 - \xi_d^1$  oraz ograniczoność  $\dot{e}$ , to prędkości  $\dot{\xi}^2(t)$  w układzie z zamkniętą pętlą sprzężenia zwrotnego będą ograniczone ( $\dot{\xi}^2(\cdot) \in \mathbb{B}$ ). Jeżeli dodatkowo  $\dot{\xi}_d^1(\cdot) \in \mathbb{L}_1$  oraz algorytm sterowania adaptacyjnego zapewnia stabilność eksponencjalną, to współrzędne  $\xi^2(t)$  będą ograniczone prawie wszędzie ( $\xi^2(\cdot) \in \mathbb{L}_\infty$ )\*.*

#### Przykład 11.3.5 (Algorytm Slotine’a-Li dla robota mobilnego)

Zastosujmy do modelu kołowego robota mobilnego we współrzędnych linearyzujących klasyczny algorytm sterowania adaptacyjnego Slotine’a-Li, który zapewnia stabilność asymptotyczną błędu śledzenia. Algorytm ten ma postać

$$\mathbf{u} = \mathbf{B}^{-1}(\xi) \left( \hat{\mathbf{J}}(\xi) \ddot{\xi}_r^1 + \hat{\mathbf{C}}(\xi, \dot{\xi}^1) \dot{\xi}_r^1 + \hat{\mathbf{D}}(\xi) - \mathbf{K}_d \mathbf{s} \right), \quad (11.83)$$

\*Zobacz dodatek A.2.

gdzie

$$\begin{aligned}
 \hat{\mathbf{J}}(\boldsymbol{\xi}) &= \mathbf{J}_0(\boldsymbol{\xi}) + \sum_{i=1}^p \mathbf{J}_i(\boldsymbol{\xi}) \hat{\boldsymbol{\theta}}_i, \\
 \hat{\mathbf{C}}(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}^1) &= \mathbf{C}_0(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}^1) + \sum_{i=1}^p \mathbf{C}_i(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}^1) \hat{\boldsymbol{\theta}}_i, \\
 \hat{\mathbf{D}}(\boldsymbol{\xi}) &= \mathbf{D}_0(\boldsymbol{\xi}) + \sum_{i=1}^p \mathbf{D}_i(\boldsymbol{\xi}) \hat{\boldsymbol{\theta}}_i, \\
 \mathbf{K}_d > \mathbf{0} &\text{ — diagonalna macierz wzmocnienia rozmiaru } m \times m, \\
 \boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_p)^T \in \mathbb{R}^p &\text{ — wektor nieznanych parametrów barycentrycznych,} \\
 \hat{\boldsymbol{\theta}} = (\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_p)^T \in \mathbb{R}^p &\text{ — wektor estymowanych wartości parametrów barycentrycznych,} \\
 \tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta} &\text{ — błąd estymacji,} \\
 \mathbf{e} = \boldsymbol{\xi}^1 - \boldsymbol{\xi}_d^1 \in \mathbb{R}^m &\text{ — błąd śledzenia,} \\
 \dot{\boldsymbol{\xi}}_r^1 = \dot{\boldsymbol{\xi}}_d^1 - \boldsymbol{\Lambda} \mathbf{e} &\text{ — prędkość zmian sygnału odniesienia,} \\
 \mathbf{s} = \dot{\boldsymbol{\xi}}^1 - \dot{\boldsymbol{\xi}}_r^1 = \dot{\mathbf{e}} + \boldsymbol{\Lambda} \mathbf{e} \in \mathbb{R}^m &\text{ — zmienna ślizgu,} \\
 \boldsymbol{\Lambda} = \boldsymbol{\Lambda}^T > \mathbf{0} &\text{ — macierz rozmiaru } m \times m.
 \end{aligned}$$

Parametry barycentryczne są estymowane stosownie do zależności

$$\hat{\boldsymbol{\theta}} = -\boldsymbol{\Gamma}^{-1} \mathbf{Y}^T(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}^1, \dot{\boldsymbol{\xi}}_r^1, \ddot{\boldsymbol{\xi}}_r^1) \mathbf{s}, \quad (11.84)$$

w której  $i$ -ta kolumna macierzy  $\mathbf{Y}(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}^1, \dot{\boldsymbol{\xi}}_r^1, \ddot{\boldsymbol{\xi}}_r^1)$ ,  $i = 1, 2, \dots, p$  jest zdefiniowana jako

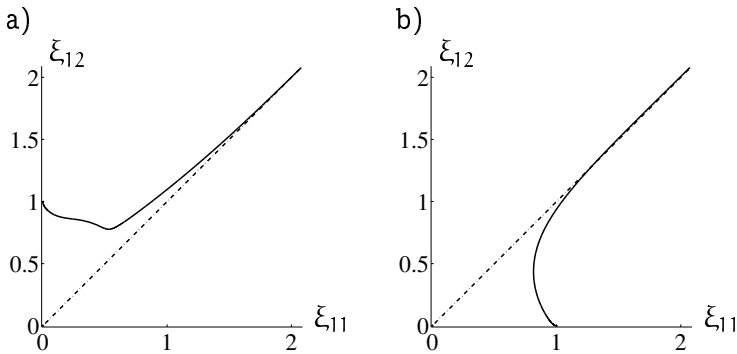
$$\mathbf{Y}(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}^1, \dot{\boldsymbol{\xi}}_r^1, \ddot{\boldsymbol{\xi}}_r^1)_{i \text{ kol}} = \mathbf{J}_i(\boldsymbol{\xi}) \ddot{\boldsymbol{\xi}}_r^1 + \mathbf{C}_i(\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}^1) \dot{\boldsymbol{\xi}}_r^1 + \mathbf{D}_i(\boldsymbol{\xi}),$$

a  $\boldsymbol{\Gamma}$  jest macierzą wzmocnienia algorytmu estymacji, rozmiaru  $p \times p$ , symetryczną i dodatnio określoną. Równania układu (11.76)–(11.77) z zamkniętą pętlą sprzężenia zwrotnego możemy przedstawić w postaci

$$\begin{cases}
 \mathbf{J}(\mathbf{e} + \boldsymbol{\xi}_d^1, \boldsymbol{\xi}^2) \dot{\mathbf{s}} = \mathbf{Y}(\mathbf{e} + \boldsymbol{\xi}_d^1, \mathbf{s} - \boldsymbol{\Lambda} \mathbf{e} + \dot{\boldsymbol{\xi}}_d^1, \boldsymbol{\xi}^2, \dot{\boldsymbol{\xi}}_d^1 - \boldsymbol{\Lambda} \mathbf{e}, \ddot{\boldsymbol{\xi}}_d^1 - \boldsymbol{\Lambda} \mathbf{s} + \boldsymbol{\Lambda}^2 \mathbf{e}) \tilde{\boldsymbol{\theta}} + \\
 \quad \quad \quad - \mathbf{C}(\mathbf{e} + \boldsymbol{\xi}_d^1, \mathbf{s} - \boldsymbol{\Lambda} \mathbf{e} + \dot{\boldsymbol{\xi}}_d^1, \boldsymbol{\xi}^2) \mathbf{s} - \mathbf{K}_d \mathbf{s} \\
 \dot{\mathbf{e}} = \mathbf{s} - \boldsymbol{\Lambda} \mathbf{e} \\
 \dot{\tilde{\boldsymbol{\theta}}} = -\boldsymbol{\Gamma}^{-1} \mathbf{Y}^T(\mathbf{e} + \boldsymbol{\xi}_d^1, \mathbf{s} - \boldsymbol{\Lambda} \mathbf{e} + \dot{\boldsymbol{\xi}}_d^1, \boldsymbol{\xi}^2, \dot{\boldsymbol{\xi}}_d^1 - \boldsymbol{\Lambda} \mathbf{e}, \ddot{\boldsymbol{\xi}}_d^1 - \boldsymbol{\Lambda} \mathbf{s} + \boldsymbol{\Lambda}^2 \mathbf{e}) \mathbf{s} \\
 \dot{\boldsymbol{\xi}}^2 = \mathbf{H}(\mathbf{e} + \boldsymbol{\xi}_d^1, \boldsymbol{\xi}^2) (\mathbf{s} - \boldsymbol{\Lambda} \mathbf{e} + \dot{\boldsymbol{\xi}}_d^1).
 \end{cases} \quad (11.85)$$

Do badania stabilności nieautonomicznego układu dynamicznego (11.85) wykorzystamy funkcję Lapunowa

$$V(\mathbf{t}, \mathbf{e}, \mathbf{s}, \boldsymbol{\xi}^2, \tilde{\boldsymbol{\theta}}) = \frac{1}{2} \mathbf{s}^T \mathbf{J}(\mathbf{e} + \boldsymbol{\xi}_d^1, \boldsymbol{\xi}^2) \mathbf{s} + \frac{1}{2} \tilde{\boldsymbol{\theta}}^T \boldsymbol{\Gamma} \tilde{\boldsymbol{\theta}}. \quad (11.86)$$



**Rysunek 11.9** Zachowanie robota mobilnego Ulisses przy zastosowaniu algorytmu Slotine'a-Li do śledzenia zadanej trajektorii położenia (11.60) we współrzędnych linearyzujących.

Równania układu w zamkniętej pętli sprzężenia zwrotnego i własność skośnej symetrii macierzy  $\mathbf{J}-2\mathbf{C}$  pozwalają wyrazić pochodną funkcji Lapunowa wzdłuż trajektorii układu (11.85) w postaci

$$\dot{V}(t, \mathbf{e}, \mathbf{s}, \xi^2, \tilde{\boldsymbol{\theta}}) = -\mathbf{s}^T \mathbf{K}_d \mathbf{s} \leq 0. \quad (11.87)$$

Warunek (11.87) gwarantuje ograniczoność zmiennych  $\mathbf{s}$  i  $\tilde{\boldsymbol{\theta}}$ , a w konsekwencji  $\mathbf{e}$  i  $\dot{\mathbf{e}}$ . Co więcej, na mocy twierdzenia La Salle'a-Yoshizawy<sup>†</sup>  $\mathbf{s}(t) \rightarrow 0$ , przy  $t \rightarrow +\infty$ , co ostatecznie pozwala wyciągnąć wniosek, że

$$\lim_{t \rightarrow +\infty} \mathbf{e}(t) = \mathbf{0}.$$

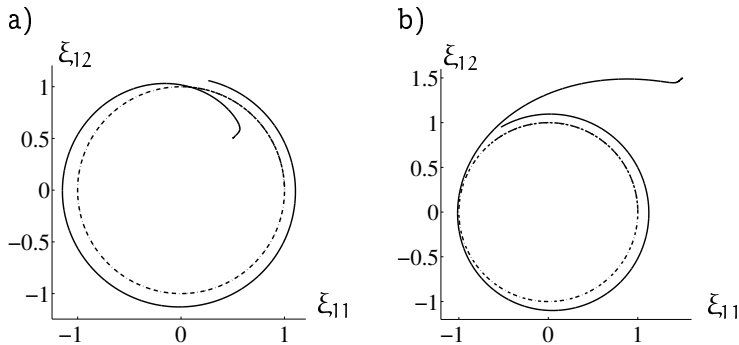
Wreszcie, na podstawie ostatniego z równań układu (11.85) wyciągamy wniosek, że ograniczoność macierzy  $\mathbf{H}(\boldsymbol{\xi})$  oraz prędkości  $\dot{\mathbf{e}}$  i  $\dot{\xi}_d^1$  zapewnia ograniczoność prędkości  $\dot{\xi}^2$ . ■

### Przykład 11.3.6 (Robot mobilny Ulisses)

Algorytm sterowania adaptacyjnego Slotine'a-Li zastosowaliśmy do modelu kołowego robota mobilnego Ulisses. Przedmiotem badań symulacyjnych było zachowanie robota przy śledzeniu dwóch trajektorii zadanych równaniami (11.60), (11.61). Przykładowe wyniki symulacji prezentujemy na rysunkach 11.9 i 11.10. Wyniki badań potwierdzają asymptotyczną zbieżność błędu śledzenia  $\mathbf{e}$ . Śledzenie zadanej trajektorii odbywa się przy ograniczonej prędkości zmian orientacji robota. ■

<sup>†</sup>Zobacz dodatek B.2.





**Rysunek 11.10** Zachowanie robota mobilnego Ulisses przy zastosowaniu algorytmu Slotine'a-Li do śledzenia zadanej trajektorii położenia (11.61) we współrzędnych linearyzujących.

### 11.3.6 Uniwersalny adaptacyjny $\lambda$ -śledzący algorytm sterowania

Przyjmijmy obecnie założenie, że model dynamiki kołowego robota mobilnego (11.77) jest całkowicie nieznan. Jak wiadomo, w tego typu sytuacjach znajdują zastosowanie układy sterowania zapewniające  $\lambda$ -śledzenie trajektorii zadanej\*. Dlatego w niniejszym podrozdziale rozważymy możliwość wykorzystania algorytmu  $\lambda$ -śledzenia do rozwiązania zadania śledzenia trajektorii zadanej dla kołowych robotów mobilnych. Naszym celem będzie rozwiązanie następującego zadania sterowania:

Znaleźć prawo sterowania w układzie sterowania (11.76)–(11.78), które zapewnia śledzenie trajektorii zadanej  $\xi_d^1(t)$  z błędem nie większym niż ustalona wartość  $\lambda > 0$  i gwarantuje ograniczoność prędkości  $\xi^2(t)$ .

Możliwość rozwiązania tego zadania za pośrednictwem algorytmu  $\lambda$ -śledzenia wynika z następującego twierdzenia.

**Twierdzenie 11.3.12** Niech  $\xi_d^1(t)$  oznacza trajektorię zadaną robota mobilnego, taką że  $\xi_d^1(t)$ ,  $\dot{\xi}_d^1(t)$  i  $\ddot{\xi}_d^1(t)$  są ciągłe i ograniczone. Wybierzmy pewną liczbę  $\lambda > 0$ . Jeżeli do układu (11.76)–(11.77) zastosujemy algorytm sterowania

$$\mathbf{u}(t) = -\mathbf{B}^{-1}(\xi)k(t)\mathbf{E}(t), \quad (11.88)$$

\*Zobacz podrozdział 6.3.3.

gdzie  $\mathbf{E}(t) = P_1 \mathbf{e}_1(t) + P_2 \mathbf{e}_2(t)$ ,  $P_1, P_2 > 0$  oraz  $\mathbf{e}_1(t) = \boldsymbol{\xi}^1(t) - \boldsymbol{\xi}_d^1(t)$ ,  $\mathbf{e}_2(t) = \dot{\boldsymbol{\xi}}^1(t) - \dot{\boldsymbol{\xi}}_d^1(t)$ , wraz z regułą adaptacji wzmocnienia  $k(t)$

$$\dot{k} = d_\lambda(\mathbf{E}) \|\mathbf{E}\| = \begin{cases} (\|\mathbf{E}\| - \lambda) \|\mathbf{E}\|, & \text{jeżeli } \|\mathbf{E}\| \geq \lambda, \\ 0, & \text{jeżeli } \|\mathbf{E}\| < \lambda, \end{cases} \quad (11.89)$$

dla  $k(0) > 0$  i dowolnego stanu początkowego  $\boldsymbol{\xi}(0) \in \mathbb{R}^n$ , to wówczas trajektoria  $(\mathbf{e}_1(t), \mathbf{e}_2(t), k(t))$  układu z zamkniętą pętlą sprzężenia zwrotnego, opisanego równaniami

$$\begin{cases} \mathbf{J}(\boldsymbol{\xi}) \dot{\mathbf{e}}_2 = -\mathbf{C}(\boldsymbol{\xi}, \mathbf{v}) \mathbf{e}_2 - k P_1 \mathbf{e}_1(t) - k P_2 \mathbf{e}_2(t) + \\ \quad \quad \quad - \mathbf{J}(\boldsymbol{\xi}) \ddot{\boldsymbol{\xi}}_d^1 - \mathbf{C}(\boldsymbol{\xi}, \mathbf{v}) \dot{\boldsymbol{\xi}}_d^1 - \mathbf{D}(\boldsymbol{\xi}) \\ \dot{\mathbf{e}}_1 = \mathbf{e}_2 \\ \dot{k} = d_\lambda(\mathbf{E}) \|\mathbf{E}\| \\ \dot{\boldsymbol{\xi}}^2 = \mathbf{H}(\boldsymbol{\xi}) (\mathbf{e}_2 + \dot{\boldsymbol{\xi}}_d^1), \end{cases} \quad (11.90)$$

posiada następujące własności:

1.  $\mathbf{E}(t) = (P_1 \mathbf{e}_1(t) + P_2 \mathbf{e}_2(t)) \rightarrow \bar{\mathbf{B}}_\lambda(0)$ , gdy  $t \rightarrow +\infty$ , gdzie  $\bar{\mathbf{B}}_\lambda(0)$  jest kulą domkniętą o środku w punkcie  $\mathbf{0}$  i promieniu  $\lambda$ ,
2.  $\lim_{t \rightarrow +\infty} k(t) < \infty$ ,
3.  $\dot{\boldsymbol{\xi}}^2(\cdot) \in \mathbb{B}$ ,

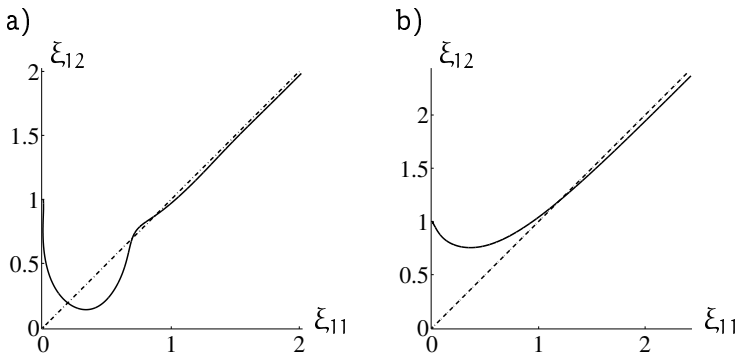
o ile tylko jest spełnione założenie  $\mathbf{e}_2(\cdot) \in \mathbb{B}$ .

Twierdzenie 11.3.12 pozwala na sformułowanie wniosku, że jeżeli wzmocnienia w pętli sprzężenia zwrotnego są adaptowane zgodnie z regułą (11.89), to wzmocnienia te i prędkości  $\dot{\boldsymbol{\xi}}^2(t)$  są ograniczone w czasie trwania regulacji, a błąd śledzenia  $\mathbf{E}(t)$  dąży do kuli o zadanym promieniu  $\lambda$ .

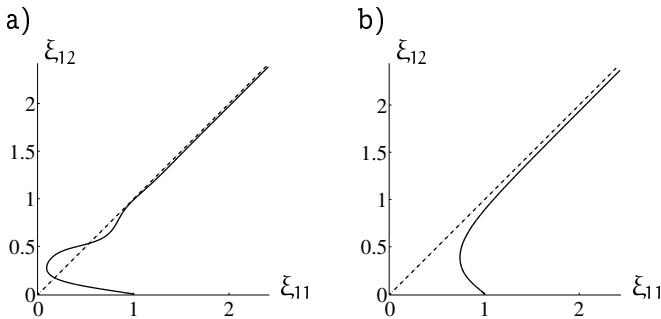
### Przykład 11.3.7 (Robot mobilny Ulisses)

Przykładowe zachowanie kołowego robota mobilnego przy zastosowaniu algorytmu  $\lambda$ -śledzenia przedstawiliśmy na rysunkach 11.11 i 11.12 oraz 11.13 i 11.14. Wartość początkowa wzmocnienia  $k(0) = 2$ , wartość parametru  $\lambda$  ustaliliśmy na poziomie 0.01. Przeprowadzone badania symulacyjne potwierdziły przydatność uniwersalnego adaptacyjnego  $\lambda$ -śledzącego algorytmu sterowania do rozwiązania zadania śledzenia trajektorii współrzędnych linearyzujących kołowych robotów mobilnych. ■

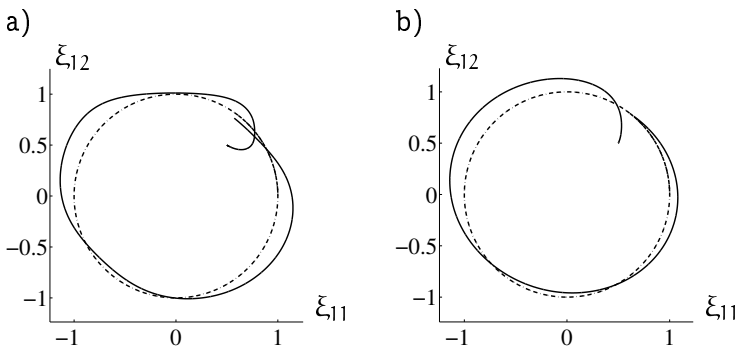
Algorytm  $\lambda$ -śledzenia nie wymaga znajomości modelu dynamiki układu, a jedynie modelu jego kinematyki (macierzy  $\mathbf{H}(\boldsymbol{\xi})$ ) oraz macierzy oddziały-



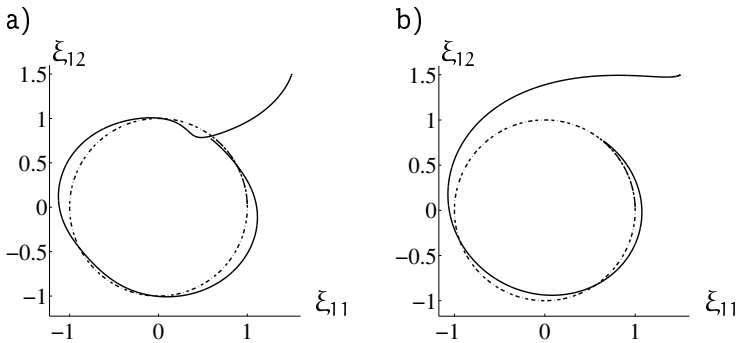
**Rysunek 11.11** Zachowanie robota mobilnego Ulisses przy zastosowaniu algorytmu  $\lambda$ -śledzenia do śledzenia zadanej trajektorii położenia (11.60) we współrzędnych linearyzujących: a)  $P_1 = 5$ ,  $P_2 = 2$ , b)  $P_1 = 5$ ,  $P_2 = 10$ .



**Rysunek 11.12** Zachowanie robota mobilnego Ulisses przy zastosowaniu algorytmu  $\lambda$ -śledzenia do śledzenia zadanej trajektorii położenia (11.60) we współrzędnych linearyzujących: a)  $P_1 = 5$ ,  $P_2 = 2$ , b)  $P_1 = 5$ ,  $P_2 = 10$ .



**Rysunek 11.13** Zachowanie robota mobilnego Ulisses przy zastosowaniu algorytmu  $\lambda$ -śledzenia do śledzenia zadanej trajektorii położenia (11.61) we współrzędnych linearyzujących: a)  $P_1 = 5$ ,  $P_2 = 2$ , b)  $P_1 = 5$ ,  $P_2 = 10$ .



**Rysunek 11.14** Zachowanie robota mobilnego Ulisses przy zastosowaniu algorytmu  $\lambda$ -śledzenia do śledzenia zadanej trajektorii położenia (11.61) we współrzędnych linearyzujących: a)  $P_1 = 5$ ,  $P_2 = 2$ , b)  $P_1 = 5$ ,  $P_2 = 10$ .

wań sterujących  $\mathbf{B}(\boldsymbol{\xi})$ . Zauważmy, że możliwość wystąpienia przeregulowań w początkowym okresie działania sterownika ogranicza możliwość jego wykorzystania w pobliżu przeszkód na scenie robota.

## 11.4 Komentarze i uwagi bibliograficzne

Dynamiki układów nieholonomicznych dotyczą prace [NF71] oraz [VG94]. Równania dynamiki układu nieholonomicznego wyprowadzone w podrozdziale 11.1 są konsekwencją Zasady d'Alemberta. Alternatywne i nierównoważne równania można uzyskać w ramach tzw. mechaniki wakonomicznej, opartej na Zasadzie Najmniejszego Działania Hamiltona [AKN88]. Na przełomie lat osiemdziesiątych i dziewięćdziesiątych pojawiło się w literaturze wiele prac poświęconych zagadnieniom modelowania i sterowania kołowych robotów mobilnych. Na rozwój teorii sterowania układów nieholonomicznych znaczący wpływ miała przełomowa praca Brocketta [Bro81], w której wykazano nieprzydatność statycznego sprzężenia zwrotnego od stanu do rozwiązania zadania stabilizacji tego typu układów. Problem modelowania kinematyki pojedynczych kołowych robotów mobilnych podjęto w pracach [ANBC91, MP87, Ży196], natomiast klasyfikację kinematyki kołowych robotów mobilnych wprowadzono w pracy [CBAN96]. Z tej pracy pochodzi twierdzenie 11.2.1. Metodę wyznaczania ograniczeń fazowych oraz algorytm wyznaczania modeli kinematyki dla pojedynczych i złożonych kołowych robotów mobilnych zaproponowano w pracy [Hos96a]. W wyniku dalszych prac udało się dokonać uogólnienia metody na przypadek

kołowych robotów mobilnych poruszających się po powierzchniach w przestrzeni trójwymiarowej, [Hos98a]. Algorytmy wyznaczania ograniczeń fazowych i wyznaczania modeli kinematyki i dynamiki zostały zaimplementowane w systemie obliczeń symbolicznych MATHEMATICA<sup>®</sup>, [Hos96b, Hos96a, Hos98b]. Wzmoczone zainteresowanie układami nieholonomicznymi zbiegło się w czasie z okresem intensywnego rozwoju geometrycznej teorii sterowania, która dostarczyła efektywnych narzędzi do badań nad specyficznymi własnościami tych układów, [BCR90, NS90, CBAN96]. Możliwość stabilizacji układów nieholonomicznych przez zmienne w czasie statyczne sprzężenie zwrotne od stanu została udowodniona przez Corona, [Cor92]. Sposób projektowania algorytmu stabilizacji dla podklasy bezdryfowych układów sterowania zaproponował Pomet w pracy [Pom92], w której podano dowód twierdzenia 11.3.7. Próbę eliminacji istotnych wad tego typu algorytmów (generowanie „dziwnych” trajektorii oraz słaba zbieżność) podjęto w pracy [PTBC92] (sterowanie hybrydowe) oraz w pracy [MM97] (sterowanie niegładkie). Zastosowanie statycznego i dynamicznego linearyzującego sprzężenia zwrotnego do sterowania kołowych robotów mobilnych zostało zaproponowane w pracy [ANCB95b]. Wyprowadzenie warunków koniecznych i wystarczających linearyzacji afinicznych układów sterowania przez statyczne sprzężenie zwrotne, przedstawione przez Jakubczyka i Respondka, [JR80], należy do najbardziej znanych osiągnięć polskiej szkoły geometrycznej teorii sterowania. Równoważne warunki linearyzacji podali Hunt, Su i Meyer, [HSM82]. Niezmiennikom równoważności przez sprzężenie zwrotne nieliniowych układów sterowania zostały poświęcone artykuły Jakubczyka [Jak90] i [Jak98]. Metoda równoważności Cartana została zastosowana w teorii linearyzacji przez Gardnera [Gar89] i Sluisa, [Slu92]. Warunki wystarczające linearyzacji przez dynamiczne sprzężenie zwrotne, zawierające koncepcję wyjść linearyzujących, przedstawiono w pracy [CLM91]. Na tej pracy opiera się sformułowanie zadania linearyzacji dynamicznej przytoczone w podrozdziale 11.3.2. Teoria równoważności dynamicznej i linearyzacji dynamicznej, wywodząca się z idei algebry różniczkowej, pochodzi od Fliessa i współpracowników, [Fli90, F<sup>+</sup>95] (układy różniczkowo płaskie) oraz Jakubczyka, [Jak92]. W artykule [FLMR99] rozszerzono pojęcia równoważności dynamicznej do równoważności orbitalnej, dopuszczającej różne skale czasu w układach równoważnych, i zdefiniowano tzw. układy orbitalnie płaskie. Dynamiczna linearyzowalność modelu kinematyki złożonego kołowego robota mobilnego została udowodniona w pracach [R<sup>+</sup>93] i [CJ94]. Wykorzystanie metody wyjść linearyzujących do zadania śledzenia trajek-

torii położenia, przy pełnej znajomości modelu dynamiki robota mobilnego, zostało omówione w pracy [ANCB95a], z której pochodzi twierdzenie 11.3.10. Badania nad odpornością takiego podejścia w przypadku ruchu z poślizgiem robotów mobilnych przeprowadzono w pracy [ANCB95b]. Zadanie stabilizacji kołowych robotów mobilnych stanowi przedmiot licznych prac [SAA90, Sam91, CS92, SW93, WTSL94, CNLM94]. Z pracy [W<sup>+</sup>92] pochodzą twierdzenia 11.3.8 i 11.3.9. Tematyki adaptacyjnego sterowania kołowych robotów mobilnych dotyczą prace [CB91, DW97]. Zastosowania klasycznych algorytmów adaptacyjnych do sterowania kołowym robotem mobilnym oraz wyniki badań eksperymentalnych zawiera praca [Hen96]. Uogólnienie metody wyjść linearyzujących na przypadek, w którym parametry modelu dynamiki są nieznane, można znaleźć w pracach [CH97, Hos96a], natomiast wykorzystanie uniwersalnego adaptacyjnego układu sterowania w przypadku strukturalnej nieznanowości modelu dynamiki prostych i złożonych kołowych robotów mobilnych zaproponowano w referatach [MH97, Maz98b]. Z pracy [CH97] pochodzi twierdzenie 11.3.11 natomiast twierdzenie 11.3.12 udowodniono w [MH97]. Metodę  $\lambda$ -śledzenia współrzędnych linearyzujących zastosowano również z powodzeniem do robotów mobilnych poruszających się w przestrzeni trójwymiarowej, [Maz98a, Maz98c]. Kwestią otwartą pozostaje także rozszerzenie konstrukcji macierzy  $\mathbf{P}$  występującej w twierdzeniu 11.2.1, aby dopuszczało dowolną orientację lokalnego układu odniesienia, a także swobodę wyboru punktu charakterystycznego. Zdefiniowanie takiej macierzy umożliwiłoby automatyczne generowanie modeli kołowych robotów mobilnych w sposób niezależny od położenia i orientacji lokalnego układu odniesienia. Pełnego opracowania wymaga również zadanie modelowania kinematyki i dynamiki robota mobilnego, którego ruch zachodzi z poślizgiem kół. Również w zakresie algorytmów sterowania uwzględnienie poślizgu kół robota należy do zadań otwartych.

## Literatura

- [AKN88] V. I. Arnold, V. V. Kozlov i A. I. Neishtadt, *Mathematical aspects of classical and celestial mechanics*. W: V. I. Arnold, (red.), *Dynamical Systems III*. Springer-Verlag, Berlin, 1988.
- [ANBC91] B. d'Andréa-Novel, G. Bastin i G. Campion, *Modelling and control of non holonomic wheeled mobile robots*. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 1130–1135, Sacramento, 1991.

- [ANCB95a] B. d'Andréa-Novel, G. Campion i G. Bastin, Control of nonholonomic wheeled mobile robots by state feedback linearization. *Int. J. Robotics Research*, 14(6):543–559, 1995.
- [ANCB95b] B. d'Andréa-Novel, G. Campion i G. Bastin, Control of wheeled mobile robots not satisfying ideal velocity constraints: A singular perturbation approach. *Int. J. Robust Control*, 5:243–267, 1995.
- [BCR90] A. M. Bloch, N. H. Mc Clamroch i M. Reyhanoglu, Controllability and stabilizability properties of a nonholonomic control systems. W: *Proc. IEEE Conf. Decision Control*, vol. 2, strony 1312–1314, Honolulu, 1990.
- [Bro81] R. W. Brockett, Control theory and singular Riemannian geometry. W: P. Hinton i G. Young, (red.), *New Directions in Applied Mathematics*, strony 11–27. Springer-Verlag, New York, 1981.
- [CB91] G. Campion i G. Bastin, Adaptive control of nonholonomic mechanical systems. W: *Proc. Europ. Contr. Conf.*, strony 1334–1338, Grenoble, 1991.
- [CBAN96] G. Campion, G. Bastin i B. d'Andréa-Novel, Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. *IEEE Trans. Robotics Automat.*, 12(1):47–62, 1996.
- [CH97] G. Campion i R. Hossa, Adaptive output linearizing control of wheeled mobile robots. W: *Proc. IFAC SyRoCo Conference*, vol. 2, strony 261–266, Nantes, 1997.
- [CJ94] P. E. Crouch i B. Jakubczyk, Dynamic transformations, linearization and chains of mechanical systems. W: *Proc. IEEE Conf. Decision Control*, vol. 3, strony 2700–2705, Orlando, 1994.
- [CLM91] B. Charlett, J. Lévine i R. Marino, Sufficient conditions for dynamic state feedback linearization. *SIAM J. Contr. Opt.*, 29(1):38–57, 1991.
- [CNLM94] C. Canudas de Wit, A. D. NDoudi-Likoho i A. Micaelli, Feedback control for a train-like vehicle. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. 1, strony 14–19, San Diego, 1994.
- [Cor92] J. M. Coron, Global asymptotic stabilization for controllable systems without drifts. *Mathematics of Control, Signal and Systems*, 5:295–315, 1992.
- [CS92] C. Canudas de Wit i O. J. Sørđalen, Exponential stabilization of mobile robot with nonholonomic constraints. *IEEE Trans. Autom. Contr.*, 37:1791–1797, 1992.

- [DW97] A. W. Divilbiss i J. T. Wen, Trajectory tracking control for a car-trailer system. *IEEE Trans. Control Syst. Technology*, **5**(3):269–278, 1997.
- [F<sup>+</sup>95] M. Fliess et al., Flatness and defect on non-linear systems: Introductory theory and examples. *Int. J. Control*, **61**:1327–1361, 1995.
- [Fli90] M. Fliess, Generalized control canonical forms for linear and nonlinear dynamics. *IEEE Trans. Autom. Contr.*, **35**:994–1001, 1990.
- [FLMR99] M. Fliess, J. Lévine, P. Martin i P. Rouchon, A Lie-Bäcklund approach to equivalence and flatness of nonlinear systems. *IEEE Trans. Autom. Contr.*, **44**(5):922–937, 1999.
- [Gar89] R. B. Gardner, *The Method of Equivalence and Its Applications*. SIAM, Philadelphia, 1989.
- [Hen96] Z. Hendzel, *Sterowanie ruchem nadźnym mobilnych robotów kołowych*. Oficyna Wydawnicza Politechniki Rzeszowskiej, Rzeszów, 1996.
- [Hos96a] R. Hossa, *Modele i algorytmy sterowania kołowych robotów mobilnych*. Rozprawa doktorska, Instytut Cybernetyki Technicznej, Politechnika Wrocławska, 1996.
- [Hos96b] R. Hossa, Zastosowanie systemu obliczeń symbolicznych mathematica do wyznaczania modeli nieholonomicznych kinematyki i dynamiki kołowych robotów mobilnych. W: *Materiały 5 Krajowej Konferencji Robotyki*, vol. **1**, strony 29–37, Świeradów Zdrój, 1996.
- [Hos98a] R. Hossa, Modele nieholonomiczne kinematyki kołowych robotów mobilnych poruszających się w przestrzeni trójwymiarowej. W: *Materiały 6 Krajowej Konferencji Robotyki*, vol. **1**, strony 275–282, Świeradów Zdrój, 1998.
- [Hos98b] R. Hossa, Symbolic modelling of the kinematics for multibody wheeled mobile robots. W: *Proc. MMAR Symposium*, vol. **3**, strony 925–930, Międzyzdroje, 1998.
- [HSM82] L. R. Hunt, R. Su i G. Meyer, Design for multi-input nonlinear systems. W: R. W. Brockett, R. S. Millman i H. J. Sussmann, (red.), *Differential Geometric Control Theory*, strony 268–298. Birkhäuser, Boston, 1982.
- [Jak90] B. Jakubczyk, Equivalence and invariants of nonlinear control systems. W: H. J. Sussmann, (red.), *Differential Geometric Control Theory*, strony 177–218. M. Dekker, New York, 1990.



- [Jak92] B. Jakubczyk, Remarks on equivalence and linearization of nonlinear systems. W: *Proc. NOLCOS Symposium*, strony 393–397, Bordeaux, 1992.
- [Jak98] B. Jakubczyk, Critical Hamiltonians and feedback invariants. W: B. Jakubczyk i W. Respondek, (red.), *Geometry of Feedback and Optimal Control*, strony 219–256. M. Dekker, New York, 1998.
- [JR80] B. Jakubczyk i W. Respondek, On linearization of control systems. *Bull. Acad. Polon. Sci. Ser. Sci. Math.*, **28**:517–522, 1980.
- [Maz98a] A. Mazur, Universal adaptive  $\lambda$ -tracker for nonholonomic wheeled mobile robots moving on the plane in 3-dimensional space. W: *Proc. ICAM SPIE Symposium*, Boston, 1998.
- [Maz98b] A. Mazur, Universal adaptive tracking controller for nonholonomic multibody wheeled mobile robots. W: *Proc. MMAR Symposium*, vol. **3**, strony 937–944, Międzyzdroje, 1998.
- [Maz98c] A. Mazur, Uniwersalny adaptacyjny układ  $\lambda$ -śledzenia trajektorii dla robotów mobilnych poruszających się w przestrzeni trójwymiarowej. W: *Materiały 6 Krajowej Konferencji Robotyki*, vol. **1**, strony 115–127, Świeradów Zdrój, 1998.
- [MH97] A. Mazur i R. Hossa, Universal adaptive  $\lambda$ -tracking controller for wheeled mobile robots. W: *Proc. IFAC SyRoCo Conference*, vol. **1**, strony 33–37, Nantes, 1997.
- [MM97] R. McCloskey i R. M. Murray, Exponential stabilization of driftless nonlinear control systems via time-varying, homogeneous feedback. *IEEE Trans. Autom. Contr.*, **42**(5):614–628, 1997.
- [MP87] P. F. Muir i Neuman C. P., Kinematic modelling for feedback control of an omnidirectional wheeled mobile robots. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. **2**, strony 1172–1178, Raleigh, 1987.
- [NF71] J. Nejmark i N. Fufajew, *Dynamika układów nieholonomicznych*. PWN, Warszawa, 1971.
- [NS90] H. Nijmeijer i A. J. van der Schaft, *Nonlinear Dynamic Control Systems*. Springer-Verlag, New York, 1990.
- [Pom92] J. B. Pomet, Explicit design of time varying stabilization control laws for a class of controllable systems without drifts. *Systems & Contr. Lett.*, **18**:147–158, 1992.
- [PTBC92] J. B. Pomet, B. Thuilot, G. Bastin i G. Campion, A hybrid strategy for the feedback stabilization of nonholonomic mobile robots. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. **1**, strony 129–134, Nice, 1992.

- [R<sup>+</sup>93] P. Rouchon et al., Flatness and motion planning: The car with n-trailers. W: *Proc. Europ. Contr. Conf.*, vol. **3**, strony 1518–1522, Groningen, 1993.
- [SAA90] C. Samson i K. Ait-Abderrahim, Feedback control of nonholonomic wheeled cart in cartesian space. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. **2**, strony 1136–1141, Sacramento, 1990.
- [Sam91] C. Samson, Time-varying feedback stabilization of car-like wheeled mobile robots. *Int. J. Robotics Research*, **12**:55–64, 1991.
- [Slu92] W. M. Sluis, *Absolute Equivalence and Its Application to Control Theory*. Rozprawa doktorska, Univ. of Waterloo, Ontario, 1992.
- [SW93] O. J. Sørдалen i K. Y. Wichlund, Exponential stabilization of a car with n trailers. W: *Proc. IEEE Conf. Decision Control*, vol. **1**, strony 978–983, San Antonio, 1993.
- [VG94] A. M. Vershik i V. Ya. Gershkovich, Nonholonomic dynamical systems, geometry of distribution and variational problems. W: V. I. Arnold, (red.), *Dynamical Systems VII*, strony 1–81. Springer-Verlag, Berlin, 1994.
- [W<sup>+</sup>92] G. Walsh et al., Stabilization of trajectories for systems with nonholonomic constraints. W: *Proc. IEEE Int. Conf. Robotics Automat.*, vol. **3**, strony 1999–2004, Nice, 1992.
- [WTSL94] G. Walsh, D. Tilbury, S. Sastry i J. P. Laumond, Stabilization of trajectories for systems with nonholonomic constraints. *IEEE Trans. Autom. Contr.*, **39**(1):216–222, 1994.
- [Ży196] W. Żylski, *Kinematyka i dynamika mobilnych robotów kołowych*. Oficyna Wydawnicza Politechniki Rzeszowskiej, Rzeszów, 1996.

# Dodatki



## Dodatek A

# Podstawowe pojęcia matematyczne

Wiele specyficznych pojęć matematycznych staraliśmy się wprowadzić na bieżąco, w odpowiednich miejscach książki. Dodatek ma na celu zdefiniowanie i objaśnienie pojęć podstawowych, których zdefiniowanie w tekście mogłoby zakłócić płynność wywodu. Są to pojęcia z zakresu algebry liniowej, analizy matematycznej, teorii układów dynamicznych i teorii sterowania.

### A.1 Algebra liniowa

Niech  $\mathcal{X}$ ,  $\mathcal{Y}$  oznaczają przestrzenie liniowe nad ciałem liczb rzeczywistych  $\mathbb{R}$  z bazami  $\{\mathbf{e}_1, \dots, \mathbf{e}_n\}$ ,  $\{\mathbf{f}_1, \dots, \mathbf{f}_m\}$ . Oznaczmy symbolem  $\mathcal{A} : \mathcal{X} \rightarrow \mathcal{Y}$  przekształcenie liniowe przestrzeni  $\mathcal{X}$  w przestrzeń  $\mathcal{Y}$ . *Macierz*  $\mathbf{A}$  przekształcenia liniowego, [Gan88], nazywamy tablicę rozmiaru  $m \times n$  złożoną z liczb  $a_{jk}$ , takich że

$$\mathcal{A}\mathbf{e}_k = \sum_{j=1}^m a_{jk} \mathbf{f}_j.$$

*Macierz dołączoną* macierzy kwadratowej  $\mathbf{A}$  rozmiaru  $n \times n$  definiujemy jako

$$\text{adj } \mathbf{A} = [(-1)^{i+j} \det \mathbf{A}_{ij}]^T, \quad (\text{A.1})$$

gdzie  $\mathbf{A}_{ij}$  jest macierzą rozmiaru  $(n-1) \times (n-1)$ , powstałą z macierzy  $\mathbf{A}$  przez wykreślenie  $i$ -tego wiersza i  $j$ -tej kolumny, [HJ86].

Niech  $\mathbf{A}$  oznacza macierz prostokątną rozmiaru  $m \times n$ . Załóżmy, że istnieje macierz  $\mathbf{X}$  o  $n$  wierszach i  $m$  kolumnach, która spełnia następujące warunki, zwane warunkami Moore'a-Penrose'a:

$$\mathbf{AXA} = \mathbf{A}, \quad (\text{A.2})$$

$$\mathbf{XAX} = \mathbf{X}, \quad (\text{A.3})$$

$$(\mathbf{AX})^T = \mathbf{AX}, \quad (\mathbf{XA})^T = \mathbf{XA}. \quad (\text{A.4})$$

Macierz  $\mathbf{X}$  nazywamy *odwrotnością uogólnioną* macierzy  $\mathbf{A}$ , jeżeli jest spełniony warunek (A.2). Macierz  $\mathbf{X}$  jest *zwrotną odwrotnością uogólnioną*, jeżeli są spełnione warunki (A.3), (A.4). Macierz  $\mathbf{X}$ , która spełnia wszystkie trzy warunki (A.2)–(A.4) nosi nazwę *pseudoodwrotności* macierzy  $\mathbf{A}$  i bywa oznaczana symbolem  $\mathbf{A}^\#$ , [Nak91].

Niech  $\mathbf{A}$  oznacza macierz kwadratową rozmiaru  $n \times n$ . *Normą macierzową* nazywamy funkcję  $\|\cdot\|$  spełniającą standardowe aksjomaty normy

$$\|\mathbf{A}\| \geq 0,$$

$$\|\mathbf{A}\| = 0 \iff \mathbf{A} = \mathbf{0},$$

$$\|c\mathbf{A}\| = |c|\|\mathbf{A}\|, \quad c \in \mathbb{R},$$

$$\|\mathbf{A} + \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|,$$

oraz tzw. aksjomat pierścienia

$$\|\mathbf{AB}\| \leq \|\mathbf{A}\|\|\mathbf{B}\|.$$

Normą macierzową indukowaną przez normę  $\|\cdot\|$  w  $\mathbb{R}^n$  nazywamy normę

$$\|\mathbf{A}\| = \max_{\|\mathbf{x}\|=1} \|\mathbf{Ax}\|.$$

Szczególnym przykładem normy macierzowej indukowanej jest *norma widmowa*, indukowana przez normę euklidesową w  $\mathbb{R}^n$ . Norma widmowa ma postać  $\|\mathbf{A}\| = \sqrt{\bar{\lambda}_{\mathbf{A}^T\mathbf{A}}}$ , gdzie przez  $\bar{\lambda}_{\mathbf{M}}$  rozumiemy największą wartość własną symetrycznej macierzy  $\mathbf{M}$ , [HJ86, Kac98].

Przy badaniu stabilności algorytmów sterowania ważną rolę odgrywa następujące

**Twierdzenie A.1.1 (Rayleigha-Ritza, [HJ86])** *Niech  $\mathbf{A}$  będzie symetryczną macierzą kwadratową. Oznaczmy przez  $\underline{\lambda}_{\mathbf{A}}$ ,  $\bar{\lambda}_{\mathbf{A}}$  najmniejszą i największą wartość własną macierzy  $\mathbf{A}$ . Wówczas prawdziwa jest nierówność*

$$\underline{\lambda}_{\mathbf{A}}\mathbf{x}^T\mathbf{x} \leq \mathbf{x}^T\mathbf{Ax} \leq \bar{\lambda}_{\mathbf{A}}\mathbf{x}^T\mathbf{x}.$$

## A.2 Funkcje i odwzorowania

Funkcja  $f: \mathbb{R}^n \rightarrow \mathbb{R}$  jest *ograniczona*, jeżeli zbiór wartości funkcji  $f(\mathbb{R}^n)$  zawiera się w pewnym skończonym przedziale  $I \subseteq \mathbb{R}$ . Funkcję  $f: \mathbb{R}^n \rightarrow \mathbb{R}$  nazywamy *gładką*, jeżeli pochodne cząstkowe rzędu  $k$  postaci  $\frac{\partial^k f}{\partial x_{i_1} \partial x_{i_2} \dots \partial x_{i_k}}(\mathbf{x})$  istnieją i są ciągłe dla dowolnego  $k$ . Funkcja  $f$  jest *analityczna*, jeżeli w pewnym otoczeniu każdego  $\mathbf{x} \in \mathbb{R}^n$  można ją rozwinąć w zbieżny szereg Taylora. Odwzorowanie  $\mathbf{f}: \mathbb{R}^n \rightarrow \mathbb{R}^m$ ,  $\mathbf{f} = (f_1, \dots, f_m)^T$  nazywamy, odpowiednio, *ograniczonym*, *gładkim* lub *analitycznym*, jeżeli wszystkie funkcje składowe  $f_i$  są ograniczone, gładkie lub analityczne. Klasę ograniczonych odwzorowań  $\mathbb{R}^n$  w  $\mathbb{R}^m$  oznaczamy symbolem  $\mathbb{B}(\mathbb{R}^n, \mathbb{R}^m)$ , gładkie odwzorowania  $\mathbb{R}^n$  w  $\mathbb{R}^m$  oznaczamy jako  $C^\infty(\mathbb{R}^n, \mathbb{R}^m)$ , dla odwzorowań analitycznych używamy oznaczenia  $C^\omega(\mathbb{R}^n, \mathbb{R}^m)$ . Odwzorowania o skończonej klasie gładkości, a więc takie, których pochodne cząstkowe są ciągłe do pewnego rzędu  $k$ , będziemy oznaczać symbolem  $C^k(\mathbb{R}^n, \mathbb{R}^m)$ . Dla odwzorowań określonych na pewnym przedziale,  $\mathbf{f}: [0, T] \rightarrow \mathbb{R}^n$  używamy, odpowiednio, oznaczeń  $C_n^k[0, T]$ ,  $C_n^\infty[0, T]$  oraz  $C_n^\omega[0, T]$ . Przestrzenie odwzorowań *miierzalnych* w sensie Lebesgue'a, przedziału  $[0, T]$  w  $\mathbb{R}^n$ , o skończonej normie  $\|\mathbf{f}\|_p = \left( \int_0^T \|\mathbf{f}(t)\|_p^p dt \right)^{1/p}$ ,  $p = 1, \dots, \infty$ , gdzie  $\|\mathbf{f}(t)\|_p = \left( \sum_{j=1}^n |f_j(t)|^p \right)^{1/p}$ ,  $\|\mathbf{f}(t)\|_\infty = \max_j |f_j(t)|$  oznaczamy symbolem  $L_n^p[0, T]$  lub jako  $L_p[0, T]$ , jeżeli przestrzeń wartości funkcji wynika jasno z kontekstu. Przy  $p = \infty$ ,  $\|\mathbf{f}\|_\infty = \max_{t \in [0, T]} \|\mathbf{f}(t)\|_\infty$ . Zauważmy, że przestrzeń  $L_n^2[0, T]$  jest *przestrzenią Hilberta*. Odwzorowanie  $\mathbf{f}: \mathbb{R}^n \rightarrow \mathbb{R}^m$  nosi nazwę *lokalnie lipschitzowskiego*, jeżeli dla każdego punktu w  $\mathbb{R}^n$  istnieje otoczenie  $\mathbf{U}$  oraz stała  $L$ , taka że dla  $\mathbf{x}, \mathbf{y} \in \mathbf{U}$   $\|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$ , [Mau71].

Odwzorowanie ograniczone  $\mathbf{f} \in \mathbb{B}(\mathbb{R}^n, \mathbb{R}^m)$  będziemy nazywać odwzorowaniem klasy  $\mathbb{B}$ , odwzorowanie  $\mathbf{f} \in C^k(\mathbb{R}^n, \mathbb{R}^m)$  nazywamy odwzorowaniem klasy  $C^k$  (dla  $k = 1, \dots, \infty, \omega$ ). Odwzorowanie  $\mathbf{f} \in L_n^k[0, T]$  zaliczamy do klasy  $L_k$ ,  $k = 1, \dots, \infty$ . Zauważmy, że odwzorowanie klasy  $L_\infty$  jest ograniczone prawie wszędzie.

Dla odwzorowania  $\mathbf{f}: \mathbb{R}^n \rightarrow \mathbb{R}^n$  definiujemy *odwzorowanie odwrotne*  $\mathbf{f}^{-1}$  spełniające warunek  $\mathbf{f} \circ \mathbf{f}^{-1} = \mathbf{f}^{-1} \circ \mathbf{f} = \mathbf{id}_{\mathbb{R}^n}$ . Ciągłe odwzorowanie  $\mathbf{f}$  posiadające ciągłe odwzorowanie odwrotne nazywamy *homeomorfizmem*.

*Macierzą Jacobiego* odwzorowania  $\mathbf{f}: \mathbb{R}^n \rightarrow \mathbb{R}^m$ ,  $\mathbf{f} = (f_1, \dots, f_m)^T$ , klasy  $C^k$  (przy  $k = 1, \dots, \infty, \omega$ ) nazywamy macierz pochodnych cząstkowych  $\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}) = \left[ \frac{\partial f_i}{\partial x_j}(\mathbf{x}) \right]$ . Jednym z najważniejszych twierdzeń analizy matematycznej jest *twierdzenie o funkcji odwrotnej*, [Mau71, AM78].

**Twierdzenie A.2.1 (o funkcji odwrotnej)** *Załóżmy, że macierz Jacobiego odwzorowania  $f: \mathbb{R}^n \rightarrow \mathbb{R}^n$  klasy  $C^k$  jest nieosobliwa w punkcie  $\mathbf{x}_0 \in \mathbb{R}^n$ , tzn.*

$$\text{rank } \frac{\partial f}{\partial \mathbf{x}}(\mathbf{x}_0) = n.$$

*Istnieje wówczas odwzorowanie*

$$f^{-1}: \mathbf{V} \rightarrow \mathbf{U},$$

*odwrotne do  $f^*$ , klasy  $C^k$ , określone na pewnych otoczeniach  $\mathbf{V} \ni f(\mathbf{x}_0)$  oraz  $\mathbf{U} \ni \mathbf{x}_0$ , zwane lokalnym dyfeomorfizmem  $\mathbb{R}^n$  klasy  $C^k$ . Przy  $k = \infty$  dyfeomorfizm nazywamy gładkim, przy  $k = \omega$  — analitycznym.*

W dowodach stabilności algorytmów sterowania, przy szacowaniu pochodnej funkcji Lapunowa względem czasu, często jest przydatne następujące twierdzenie o wartości średniej, [Mau71, AMR88].

**Twierdzenie A.2.2 (o wartości średniej)** *Dla funkcji  $f: \mathbb{R}^n \rightarrow \mathbb{R}$  klasy  $C^1$  zachodzi wzór*

$$f(\mathbf{x}) - f(\mathbf{x}_0) = \int_0^1 \sum_{i=1}^n (x_i - x_{0i}) \frac{\partial f}{\partial x_i} (t\mathbf{x} + (1-t)\mathbf{x}_0) dt.$$

### A.3 Układy dynamiczne, pola wektorowe, nawiasy Liego

Powiemy, że układ równań różniczkowych

$$\dot{\mathbf{x}} = \mathbf{X}(\mathbf{x}), \tag{A.5}$$

w którym  $\mathbf{X} = (X_1, \dots, X_n)^T$ ,  $\mathbf{x} \in \mathbb{R}^n$ , definiuje *układ dynamiczny* w  $\mathbb{R}^n$ , jeżeli krzywe całkowe  $\mathbf{x}(t)$  układu (A.5) są określone w każdej chwili czasu  $t \in \mathbb{R}$ . Odwzorowanie  $\mathbf{X}$  określające układ dynamiczny nazywamy *polem wektorowym*, a przestrzeń  $\mathbb{R}^n$  — *przestrzenią stanu* układu. Zakładamy, że pole  $\mathbf{X}$  jest gładkie, to znaczy  $\mathbf{X} \in C^\infty(\mathbb{R}^n, \mathbb{R}^n)$ . Każdy układ dynamiczny definiuje odwzorowanie  $\boldsymbol{\varphi}(t, \mathbf{x})$  zwane *strumieniem*. Ze względu na formalne podobieństwo do strumienia liniowego układu dynamicznego, bardzo często stosowanym oznaczeniem strumienia pola  $\mathbf{X}$  jest  $\boldsymbol{\varphi}(t, \mathbf{x}) =$

\*Tzn. takie, że  $f \circ f^{-1} = \text{id}_{\mathbf{V}}$  oraz  $f^{-1} \circ f|_{\mathbf{U}} = \text{id}_{\mathbf{U}}$ .



$\exp(t\mathbf{X})(\mathbf{x})$ . Strumień spełnia układ równań (A.5), co oznacza, że  $\frac{d}{dt}\boldsymbol{\varphi}(t, \mathbf{x}) = \mathbf{X}(\boldsymbol{\varphi}(t, \mathbf{x}))$ . Dla ustalonego stanu początkowego  $\mathbf{x}$  strumień określa krzywą całkową (trajektorię) układu dynamicznego przechodzącą przez punkt  $\mathbf{x}$ , a dla ustalonej chwili czasu  $t$  strumień wyznacza wzajemnie jednoznaczne i gładkie odwzorowanie  $\boldsymbol{\varphi}_t : \mathbb{R}^n \rightarrow \mathbb{R}^n$ , będące dyfeomorfizmem przestrzeni stanu. Podzbiór  $\Omega \subseteq \mathbb{R}^n$  nazywamy *zbiorem niezmienniczym* układu dynamicznego (A.5), jeżeli dla każdego  $\mathbf{x} \in \Omega$  trajektoria  $\boldsymbol{\varphi}(t, \mathbf{x}) \in \Omega$  dla  $t \in \mathbb{R}$ . Najprostszym przykładem zbioru niezmienniczego jest punkt równowagi zdefiniowany równaniem  $\mathbf{X}(\mathbf{x}) = \mathbf{0}$ . Punkt równowagi układu dynamicznego bywa też nazywany punktem osobliwym pola  $\mathbf{X}$ . Punkt, w którym pole wektorowe nie znika, nazywa się punktem regularnym. Zachowanie układu dynamicznego w otoczeniu punktów regularnych opisuje następujące *twierdzenie o prostowaniu*.

**Twierdzenie A.3.1 (o prostowaniu)** *Niech  $\mathbf{x} \in \mathbb{R}^n$  oznacza punkt regularny pola  $\mathbf{X}$ . Wówczas, istnieje lokalny dyfeomorfizm  $\xi = \Phi(\mathbf{x})$ , zdefiniowany na pewnym otoczeniu punktu  $\mathbf{x}$ , taki że w nowych współrzędnych pole  $\mathbf{X} = \mathbf{e}_1 \in \mathbb{R}^n$ .*

Układ dynamiczny postaci (A.5) bywa nazywany *autonomicznym* (niezależnym od czasu). W przypadku, gdy pole  $\mathbf{X}$  zależy w sposób jawny od czasu, a więc gdy

$$\dot{\mathbf{x}} = \mathbf{X}(\mathbf{x}, t), \quad (\text{A.6})$$

układ dynamiczny nosi nazwę *nieautonomicznego*. Punkt  $\mathbf{x}_0 \in \mathbb{R}^n$  nazywamy punktem równowagi nieautonomicznego układu dynamicznego, jeżeli dla każdego  $t \in \mathbb{R}$  zachodzi  $\mathbf{X}(\mathbf{x}_0, t) = \mathbf{0}$ , [Arn75, AA88, AM78].

Niech będą dane dwa pola wektorowe  $\mathbf{X}, \mathbf{Y} \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R}^n)$ . Oznaczmy symbolem  $\boldsymbol{\varphi}_t(\mathbf{x}) = \boldsymbol{\varphi}(t, \mathbf{x})$  strumień układu (A.5) (pola  $\mathbf{X}$ ) i zdefiniujemy następujące *działanie pola  $\mathbf{X}$  na pole  $\mathbf{Y}$*

$$\text{Ad}_{\mathbf{X}}^t \mathbf{Y}(\mathbf{x}) = \left( \frac{\partial \boldsymbol{\varphi}_{-t}}{\partial \mathbf{x}} \mathbf{Y} \right) \circ \boldsymbol{\varphi}_t(\mathbf{x}). \quad (\text{A.7})$$

*Nawiasem Liego* pól  $\mathbf{X}$  i  $\mathbf{Y}$  nazywamy pole wektorowe

$$[\mathbf{X}, \mathbf{Y}](\mathbf{x}) = \left. \frac{d}{dt} \text{Ad}_{\mathbf{X}}^t \mathbf{Y}(\mathbf{x}) \right|_{t=0}. \quad (\text{A.8})$$

Nietrudno wykazać, że z formuły (A.8) wynika zależność (2.34), którą można uważać za definicję nawiasu Liego we współrzędnych. Dla wygody za-

pisu iterowanych nawiasów Liego wprowadza się oznaczenie  $\text{ad}_{\mathbf{X}}^0 \mathbf{Y} = \mathbf{Y}$  oraz  $\text{ad}_{\mathbf{X}}^{k+1} \mathbf{Y} = [\mathbf{X}, \text{ad}_{\mathbf{X}}^k \mathbf{Y}]$ . Pola wektorowe z operacją nawiasu Liego tworzą tzw. *algebrę Liego*, [AM78, Gan88, Spi79, Sas99].

Niech będzie dana gładka funkcja  $H(\mathbf{q}, \mathbf{p})$  określona w  $\mathbb{R}^{2n}$ , zwana *funkcją Hamiltona* lub *hamiltonianem*. Układ dynamiczny postaci

$$\dot{\mathbf{q}} = \frac{\partial H}{\partial \mathbf{p}}(\mathbf{q}, \mathbf{p}), \quad \dot{\mathbf{p}} = -\frac{\partial H}{\partial \mathbf{q}}(\mathbf{q}, \mathbf{p}) \quad (\text{A.9})$$

nazywamy *układem hamiltonowskim*, [AM78].

Niech będzie dany gładki układ dynamiczny postaci (A.5). Gładką funkcję  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}$  nazywamy *całką pierwszą* (stałą ruchu) układu (A.4), jeżeli  $\mathbf{f}$  jest stała na trajektoriach tego układu, tzn.

$$\dot{\mathbf{f}}(\mathbf{x}) = d\mathbf{f}(\mathbf{x})\mathbf{X}(\mathbf{x}) = 0.$$

Nietrudno sprawdzić, że hamiltonian jest całką pierwszą hamiltonowskiego układu dynamicznego.

## A.4 Układy sterowania

*Układem sterowania* nazywamy obiekt opisany układem równań różniczkowych

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t), \quad (\text{A.10})$$

gdzie  $\mathbf{x} \in \mathbb{R}^n$ , zależnych od zmiennej  $\mathbf{u} \in \mathbb{R}^m$  zwanej sterowaniem. Przestrzeń  $\mathbb{R}^n$  nazywa się przestrzenią stanu układu (A.10), zaś odwzorowania  $\mathbf{u} : \mathbb{R} \rightarrow \mathbb{R}^m$  są nazywane funkcjami sterującymi. Zakłada się, że funkcje sterujące  $\mathbf{u}(\cdot)$  należą do pewnej klasy sterowań dopuszczalnych  $\mathcal{U}$ . Z reguły do klasy  $\mathcal{U}$  należą sterowania odcinkami stałe. Warunki nakładane na odwzorowanie  $\mathbf{f}$  definiujące układ sterowania i klasę sterowań dopuszczalnych powinny zapewniać istnienie trajektorii układu (A.10). Układ sterowania (A.10) jest nazywany *zależnym od czasu*, *nieautonomicznym* lub *niestacjonarnym*. Ważną podklasą układów sterowania stanowią *układy liniowe* zdefiniowane w następujący sposób

$$\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x} + \mathbf{B}(t)\mathbf{u}. \quad (\text{A.11})$$

Jeżeli  $\mathbf{f}$  nie zależy jawnie od czasu, układ sterowania jest *niezależny od czasu*, *autonomiczny* lub *stacjonarny*. Mamy wówczas

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}). \quad (\text{A.12})$$

Wśród układów niezależnych od czasu wyróżniamy podklasę *afinicznych* układów sterowania

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} = \mathbf{F}(\mathbf{x}) + \sum_{k=1}^m \mathbf{g}_k(\mathbf{x})u_k. \quad (\text{A.13})$$

Jeżeli w układzie afinicznym  $\mathbf{F}(\mathbf{x}) \equiv \mathbf{0}$ , a więc

$$\dot{\mathbf{x}} = \mathbf{G}(\mathbf{x})\mathbf{u} = \sum_{k=1}^m \mathbf{g}_k(\mathbf{x})u_k, \quad (\text{A.14})$$

uzyskujemy układ liniowy ze względu na sterowanie, zwany układem *bezdryfowym* lub *polisystemem dynamicznym*. Termin polisystem dynamiczny jest odzwierciedleniem tego, że przy sterowaniach odcinkami stałych układ (A.14) można utożsamić z rodziną pól wektorowych  $\mathbf{g}_k(\mathbf{x})$ . Układ afiniczny, w którym pole  $\mathbf{f}(\mathbf{x}) = \mathbf{Ax}$ , a pola  $\mathbf{g}_k(\mathbf{x}) = \mathbf{b}_k = \text{const}$  staje się *liniowym* układem sterowania

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu}. \quad (\text{A.15})$$

Podstawową własnością układów sterowania jest *sterowalność*. W teorii sterowania są znane różne własności typu sterowalności. Naszym potrzebom najbardziej służą własności: *sterowalność*, *lokalna sterowalność* i *sterowalność w krótkim czasie*. Mówimy, że układ sterowania jest sterowalny, jeżeli przy pomocy dopuszczalnych funkcji sterujących i w skończonym czasie z każdego stanu początkowego układu można osiągnąć każdy zadany stan w przestrzeni stanu. Układ sterowania jest lokalnie sterowalny, jeżeli zbiór stanów osiągalnych w skończonym czasie z każdego zadanego stanu  $\mathbf{x} \in \mathbb{R}^n$  przy użyciu dopuszczalnych funkcji sterujących zawiera otoczenie tego stanu. Układ sterowania jest sterowalny w krótkim czasie, gdy zbiór stanów możliwych do osiągnięcia z każdego stanu w dowolnym czasie  $t > 0$  pokrywa się z przestrzenią stanu układu oraz jest *lokalnie sterowalny w krótkim czasie*, gdy zbiór stanów osiągalnych w dowolnym czasie  $t > 0$  z każdego stanu zawiera otoczenie tego stanu.

Mówimy, że układ sterowania (A.10) ma punkt równowagi  $\mathbf{0} \in \mathbb{R}^n$ , jeżeli  $\mathbf{f}(\mathbf{0}, \mathbf{u}, t) = \mathbf{0}$  w każdej chwili  $t \in \mathbb{R}$ . Punkt równowagi nazywa się *stabilizowalnym*, jeżeli istnieje dopuszczalna funkcja sterująca  $\mathbf{u}(t)$ , taka że punkt równowagi  $\mathbf{0}$  nieautonomicznego układu dynamicznego

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}(t), t)$$

jest asymptotycznie stabilny. Rozwinięcie przedstawionych tu koncepcji można znaleźć w literaturze [Sus82, Isi89, Son90, NS90, Sas99].

## A.5 Zasada Maksimum Pontriagina

Rozważmy niezależny od czasu układ sterowania opisany równaniami

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}), \quad (\text{A.16})$$

z wektorem stanu  $\mathbf{x} \in \mathbb{R}^n$  i wektorem sterowań  $\mathbf{u} \in \mathbb{R}^m$ . Niech *zadanie sterowania* polega na przeprowadzeniu układu (A.16) ze stanu początkowego  $\mathbf{x}_0$  do stanu końcowego  $\mathbf{x}_1$  w taki sposób, by minimalizować wskaźnik jakości

$$\mathcal{J}(\mathbf{u}(\cdot)) = \int_{t_0}^{t_1} L(\mathbf{x}(t), \mathbf{u}(t)) dt. \quad (\text{A.17})$$

Wprowadźmy *zmienne dołączone*  $(\boldsymbol{\psi}, \psi_0) \in \mathbb{R}^{n+1}$ ,  $\psi_0 = \text{const}$ , i przyporządkujmy układowi (A.16) ze wskaźnikiem (A.17) hamiltonian postaci

$$H(\mathbf{x}, \boldsymbol{\psi}, \psi_0, \mathbf{u}) = \boldsymbol{\psi}^T \mathbf{f}(\mathbf{x}, \mathbf{u}) - \psi_0 L(\mathbf{x}, \mathbf{u}). \quad (\text{A.18})$$

W myśl *Zasady Maksimum Pontriagina*, [PBG60, AF69, VG97], warunkiem koniecznym optymalności sterowania  $\mathbf{u}(t)$  jest przyjmowanie przez hamiltonian wartości maksymalnej

$$H(\mathbf{x}(t), \boldsymbol{\psi}(t), \psi_0, \mathbf{u}(t)) = \max_{\mathbf{u}} H(\mathbf{x}(t), \boldsymbol{\psi}(t), \psi_0, \mathbf{u}) \quad (\text{A.19})$$

wzdłuż trajektorii  $(\mathbf{x}(t), \boldsymbol{\psi}(t))$  nieautonomicznego układu hamiltonowskiego opisanego przez *równania kanoniczne Hamiltona*

$$\begin{aligned} \dot{\mathbf{x}} &= \frac{\partial H}{\partial \boldsymbol{\psi}}(\mathbf{x}(t), \boldsymbol{\psi}(t), \psi_0, \mathbf{u}(t)), \\ \dot{\boldsymbol{\psi}} &= -\frac{\partial H}{\partial \mathbf{x}}(\mathbf{x}(t), \boldsymbol{\psi}(t), \psi_0, \mathbf{u}(t)). \end{aligned} \quad (\text{A.20})$$

Trójkę funkcji  $(\mathbf{u}(t), \mathbf{x}(t), \boldsymbol{\psi}(t))$  spełniających warunek (A.19) oraz równania (A.20) nazywamy ekstremalą zadania sterowania optymalnego. Ekstremalę odpowiadającą wartości  $\psi_0 = 0$ , wzdłuż której hamiltonian znika tożsamościowo,  $H(\mathbf{x}(t), \boldsymbol{\psi}(t), \psi_0, \mathbf{u}(t)) \equiv 0$ , nazywamy osobliwą. Istnienie ekstremal osobliwych wynika z własności układu sterowania, nie z postaci wskaźnika jakości. Ekstremale nieosobliwe noszą nazwę regularnych. Przy ich wyznaczaniu można założyć, że zmienna  $\psi_0 = 1$ .

## A.6 Równania Eulera-Lagrange'a

Niech będzie dany gładki lagranżian  $L(\mathbf{u}, \frac{\partial \mathbf{u}}{\partial \mathbf{x}}, \mathbf{x}) = L(u_1(\mathbf{x}), \dots, u_n(\mathbf{x}), u_{1x_1}(\mathbf{x}), \dots, u_{1x_m}(\mathbf{x}), \dots, u_{nx_1}(\mathbf{x}), \dots, u_{nx_m}(\mathbf{x}), x_1, \dots, x_m)$  zależny od funkcji  $u_i(\mathbf{x})$  oraz ich pochodnych cząstkowych  $u_{ix_j}(\mathbf{x}) = \frac{\partial u_i}{\partial x_j}(\mathbf{x})$ ,  $i = 1, \dots, n$ ,  $j = 1, \dots, m$ . Załóżmy, że  $\mathbf{V}$  jest pewnym obszarem w  $\mathbb{R}^m$ . Warunkiem koniecznym ekstremum funkcjonału

$$\mathcal{J}(\mathbf{u}(\cdot)) = \int_{\mathbf{V}} L(\mathbf{u}, \frac{\partial \mathbf{u}}{\partial \mathbf{x}}, \mathbf{x}) \, d\mathbf{x} \quad (\text{A.21})$$

jest spełnienie następujących równań Eulera-Lagrange'a, [GF79]:

$$\sum_{j=1}^m \frac{\partial}{\partial x_j} \frac{\partial L}{\partial u_{ix_j}} - \frac{\partial L}{\partial u_i} = 0, \quad i = 1, \dots, n. \quad (\text{A.22})$$

W przypadku, gdy funkcje  $u_i$  zależą od jednej zmiennej,  $u_i(t) = q_i(t)$ , a zatem lagranżian jest równy  $L(\mathbf{q}, \dot{\mathbf{q}}, t)$ , równania (A.22) przyjmują klasyczną postać

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\mathbf{q}}} - \frac{\partial L}{\partial \mathbf{q}} = \mathbf{0}. \quad (\text{A.23})$$

## Literatura

- [AA88] D. V. Anosov i V. I. Arnold, (red.), *Dynamical Systems I*. Springer-Verlag, Berlin, 1988.
- [AF69] M. Athans i P. L. Falb, *Sterowanie optymalne. Wstęp do teorii i jej zastosowania*. WNT, Warszawa, 1969.
- [AM78] R. Abraham i J. E. Marsden, *Foundations of Mechanics*. The Benjamin, Reading, 1978.
- [AMR88] R. Abraham, J. E. Marsden i T. Ratiu, *Manifolds, Tensor Analysis, and Applications*. Springer-Verlag, New York, 1988.
- [Arn75] V. I. Arnold, *Równania różniczkowe zwyczajne*. PWN, Warszawa, 1975.
- [Gan88] F. R. Gantmacher, *Teoria macierzy*. Nauka, Moskwa, 1988.
- [GF79] I. M. Gelfand i S. W. Fomin, *Rachunek wariacyjny*. PWN, Warszawa, 1979.
- [HJ86] R. A. Horn i C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1986.

- [Isi89] A. Isidori, *Nonlinear Control Systems*. Springer-Verlag, Berlin, 1989.
- [Kac98] T. Kaczorek, *Wektory i macierze w automatyce i elektrotechnice*. WNT, Warszawa, 1998.
- [Mau71] K. Maurin, *Analiza cz. I*. PWN, Warszawa, 1971.
- [Nak91] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*. Addison-Wesley, Reading, 1991.
- [NS90] H. Nijmeijer i A. J. van der Schaft, *Nonlinear Dynamic Control Systems*. Springer-Verlag, New York, 1990.
- [PBG60] L. S. Pontriagin, V. G. Boltianski i R. V. Gamkrelidze, Teoria optymalnych procesów. I Princip maksimuma. *Izv. Sov. AN*, **24**(1):3–42, 1960.
- [Sas99] S. S. Sastry, *Nonlinear Systems*. Springer-Verlag, New York, 1999.
- [Son90] E. D. Sontag, *Mathematical Control Theory*. Springer-Verlag, New York, 1990.
- [Spi79] M. Spivak, *A Comprehensive Introduction to Differential Geometry*. Publish or Perish, Berkeley, 1979.
- [Sus82] H. J. Sussmann, Lie brackets, real analyticity and geometric control. W: R. W. Brockett, R. S. Millman i H. J. Sussmann, (red.), *Differential Geometric Control Theory*, strony 1–116. Birkhäuser, Boston, 1982.
- [VG97] T. L. Vincent i W. J. Grantham, *Nonlinear and Optimal Control Systems*. J. Wiley and Sons, New York, 1997.

## Dodatek B

# Wybrane twierdzenia o stabilności

### B.1 Układy autonomiczne

Przyjmujemy, że definicja *stabilności*, pojęcie *funkcji Lapunowa* oraz sformułowanie *I i II Metody Lapunowa* dla autonomicznych układów dynamicznych są Czytelnikowi znane, [Dem72, SL91, Sas99]. Oprócz metod Lapunowa, do podstawowych narzędzi do badania stabilności układów autonomicznych należy *Zasada Niezmienniczości La Salle'a*, która może być wypowiedziana w postaci następującego twierdzenia.

**Twierdzenie B.1.1** (La Salle, [SL91, Sas99]) *Niech będzie dany układ dynamiczny*

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \tag{B.1}$$

$\mathbf{x} \in \mathbb{R}^n$ , z funkcją Lapunowa  $V(\mathbf{x})$  klasy  $\mathcal{C}^1$ , taką że

$$\dot{V}(\mathbf{x}) = dV(\mathbf{x})\mathbf{f}(\mathbf{x}) \leq 0.$$

*Definiujemy zbiór*

$$\mathbf{E} = \left\{ \mathbf{x} \in \mathbb{R}^n \mid \dot{V}(\mathbf{x}) = 0 \right\}.$$

*Wówczas, ograniczone trajektorie układu (B.1) dążą do największego zbioru niezmienniczego  $\mathbf{M} \subseteq \mathbf{E}$ .*

## B.2 Układy nieautonomiczne

### B.2.1 II Metoda Lapunowa

Przed sformułowaniem twierdzenia o stabilności wprowadzimy następującą definicję.

**Definicja B.2.1** *Funkcja ciągła  $\gamma : [0, a) \rightarrow \mathbb{R}_+$  należy do klasy  $\mathbb{K}$ , jeśli jest ściśle rosnąca i  $\gamma(0) = 0$ . Jeśli dodatkowo  $a = \infty$  oraz  $\gamma(r) \rightarrow \infty$  dla  $r \rightarrow \infty$ , to funkcja  $\gamma$  należy do klasy  $\mathbb{K}_\infty$ .*

**Twierdzenie B.2.1 (II Metoda Lapunowa, [KKK95])** *Niech  $\mathbf{x} = \mathbf{0}$  będzie punktem równowagi gładkiego układu nieautonomicznego\**

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t). \quad (B.2)$$

Zdefiniujmy kulę otwartą  $\mathbf{D} = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\| < r\}$ . Niech  $V : \mathbf{D} \times \mathbb{R} \rightarrow \mathbb{R}_+$  będzie funkcją klasy  $\mathbb{C}^1$ , taką że dla wszystkich  $t \geq 0$ ,  $\mathbf{x} \in \mathbf{D}$  zachodzi

$$\begin{aligned} \gamma_1(\|\mathbf{x}\|) &\leq V(\mathbf{x}, t) \leq \gamma_2(\|\mathbf{x}\|), \\ \dot{V} &= \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}, t) \leq -\gamma_3(\|\mathbf{x}\|). \end{aligned}$$

Wówczas punkt równowagi  $\mathbf{x} = \mathbf{0}$  jest:

- jednostajnie stabilny, jeśli  $\gamma_1$  i  $\gamma_2$  są funkcjami klasy  $\mathbb{K}$  na przedziale  $[0, r)$  oraz  $\gamma_3(\cdot) \geq 0$  na przedziale  $[0, r)$ ,
- jednostajnie asymptotycznie stabilny, jeśli  $\gamma_1$ ,  $\gamma_2$  i  $\gamma_3$  są funkcjami klasy  $\mathbb{K}$  na przedziale  $[0, r)$ ,
- eksponencjalnie stabilny, jeśli  $\gamma_i(\rho) = k_i \rho^\alpha$  na przedziale  $[0, r)$ , przy  $k_i > 0$ ,  $\alpha > 0$ ,  $i = 1, 2, 3$ ,
- globalnie jednostajnie stabilny, jeśli  $\mathbf{D} = \mathbb{R}^n$ ,  $\gamma_1$  i  $\gamma_2$  są funkcjami klasy  $\mathbb{K}_\infty$ , a  $\gamma_3(\cdot) \geq 0$  na  $\mathbb{R}_+$ ,
- globalnie jednostajnie asymptotycznie stabilny, jeśli  $\mathbf{D} = \mathbb{R}^n$ , funkcje  $\gamma_1$  i  $\gamma_2$  są klasy  $\mathbb{K}_\infty$ , a  $\gamma_3$  jest funkcją klasy  $\mathbb{K}$  na  $\mathbb{R}_+$ ,

---

\*W istocie wystarczy, żeby odwzorowanie  $\mathbf{f} : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$  było lokalnie lipschitzowskie względem  $\mathbf{x}$  i ciągłe względem  $t$ .



- globalnie eksponencjalnie stabilny, jeśli  $\mathbf{D} = \mathbb{R}^n$ , a  $\gamma_i(\rho) = k_i \rho^\alpha$  na  $\mathbb{R}_+$ ,  $k_i > 0$ ,  $\alpha > 0$ ,  $i = 1, 2, 3$ .

## B.2.2 Lemat Barbalata

**Lemat B.2.1** (Barbalat, [SL91]) *Niech  $f(t)$  będzie pewną funkcją rzeczywistą klasy  $\mathbb{C}^2$ . Załóżmy, że  $f(t)$  jest ograniczona od dołu, oraz że pochodna  $\dot{f}(t) \leq 0$ . Wówczas, jeżeli druga pochodna  $\ddot{f}(t)$  jest ograniczona ( $\ddot{f}(\cdot) \in \mathbb{B}$ ), to  $\lim_{t \rightarrow +\infty} \dot{f}(t) = 0$ .*

## B.2.3 Twierdzenie La Salle'a-Yoshizawy

Odpowiednikiem Zasady Niezmienniczości La Salle'a dla układów nieautonomicznych jest następujące *twierdzenie La Salle'a-Yoshizawy*.

**Twierdzenie B.2.2** (La Salle-Yoshizawa, [KKK95, Sas99]) *Niech  $\mathbf{x} = \mathbf{0}$  będzie punktem równowagi układu nieautonomicznego (B.2). Załóżmy, że  $V : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$  jest funkcją klasy  $\mathbb{C}^1$ , taką że dla każdego czasu  $t \geq 0$ ,  $\mathbf{x} \in \mathbb{R}^n$*

$$\begin{aligned} \gamma_1(\|\mathbf{x}\|) &\leq V(\mathbf{x}, t) \leq \gamma_2(\|\mathbf{x}\|), \\ \dot{V} &= \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}, t) \leq -W(\mathbf{x}) \leq 0, \end{aligned}$$

gdzie  $\gamma_1, \gamma_2$  są funkcjami klasy  $K_\infty$ , zaś  $W$  jest funkcją ciągłą. Wówczas, wszystkie trajektorie  $\mathbf{x}(t)$  układu (B.2) są globalnie ograniczone i spełniają warunek

$$\lim_{t \rightarrow +\infty} W(\mathbf{x}(t)) = 0.$$

Jeśli dodatkowo funkcja  $W(\mathbf{x})$  jest dodatnio określona, to punkt  $\mathbf{x} = \mathbf{0}$  jest globalnie jednostajnie asymptotycznie stabilnym punktem równowagi.

## B.2.4 Lemat Wena-Bayarda

Jednym z ważnych lokalnych twierdzeń używanych do dowodzenia stabilności algorytmów sterowania jest lemat podany przez Wena i Bayarda.

**Lemat B.2.2 (Wen-Bayard, [WB88])** Niech będzie dany nieautonomiczny układ dynamiczny opisany równaniami

$$\dot{x}_i = f_i(x_1, \dots, x_N, t),$$

gdzie, dla każdego  $i = 1, \dots, N$ ,  $x_i \in \mathbb{R}^{n_i}$  oraz  $t \geq 0$ . Zakładamy, że odwzorowania  $f_i$  są lokalnie lipschitzowskie ze względu na  $x_1, \dots, x_N$ , jednostajnie względem  $t$  na przedziałach ograniczonych, oraz ciągłe względem  $t$  dla  $t \geq 0$ . Niech będzie dana funkcja  $V: \mathbb{R}^{(n_1+n_2+\dots+n_N)} \times \mathbb{R} \rightarrow \mathbb{R}$  postaci

$$V(x_1, \dots, x_N, t) = \sum_{i,j=1}^N x_i^T P_{ij}(x_1, \dots, x_N, t) x_j, \quad (B.3)$$

taka że dla każdego  $i = 1, \dots, N$  istnieje liczba  $\xi_i > 0$ , dla której

$$\xi_i \|x_i\|^2 \leq V(x_1, \dots, x_N, t) \quad (B.4)$$

oraz

$$\dot{V}(x_1, \dots, x_N, t) \leq - \sum_{i \in I_1} \left( \alpha_i - \sum_{j \in I_{2_i}} \gamma_{ij} \|x_j(t)\|^{k_{ij}} \right) \|x_i(t)\|^2, \quad (B.5)$$

gdzie  $\alpha_i, \gamma_{ij}, k_{ij} > 0$ ,  $I_{2_i} \subset I_1 \subset \{1, \dots, N\}$ . Założmy, że w chwili początkowej  $V(x_1(0), \dots, x_N(0), 0) = V_0$ . Jeśli dla każdego  $i \in I_1$  zachodzi

$$\alpha_i > \sum_{j \in I_{2_i}} \gamma_{ij} \left( \frac{V_0}{\xi_j} \right)^{\frac{k_{ij}}{2}},$$

to dla każdego

$$\lambda_i \in \left( 0, \alpha_i - \sum_{j \in I_{2_i}} \gamma_{ij} \left( \frac{V_0}{\xi_j} \right)^{\frac{k_{ij}}{2}} \right)$$

jest spełniona nierówność

$$\dot{V}(x_1, \dots, x_N, t) \leq - \sum_{i \in I_1} \lambda_i \|x_i\|^2, \quad \forall t \geq 0.$$

## Literatura

- [Dem72] B. P. Demidowicz, *Matematyczna teoria stabilności*. WNT, Warszawa, 1972.

- 
- [KKK95] M. Krstić, I. Kanellakopoulos i P. Kokotović, *Nonlinear and Adaptive Control Design*. J. Wiley and Sons, New York, 1995.
- [Sas99] S. S. Sastry, *Nonlinear Systems*. Springer-Verlag, New York, 1999.
- [SL91] J. J. E. Slotine i W. Li, *Applied Nonlinear Control*. Prentice Hall, Englewood Cliffs, 1991.
- [WB88] J. T. Wen i D. S. Bayard, New class of control laws for robotic manipulators: Part 1 — non-adaptive case. *Int. J. Control*, **47**(5):1361–1386, 1988.



## Dodatek C

# Wybrane dowody stabilności algorytmów sterowania

Ostatni dodatek poświęcimy przedstawieniu wybranych dowodów stabilności algorytmów sterowania. Dowód stabilności algorytmu Sadegha-Horowitza jest typowym dowodem stabilności eksponencjalnej przeprowadzonym w oparciu o drugą metodę Lapunowa. Podobny sposób rozumowania można zastosować w przypadku algorytmu Slotine'a-Li (w części dotyczącej zbieżności zmiennej ślizgu) oraz algorytmu uniwersalnego. Dowód stabilności algorytmu adaptacyjnego Slotine'a-Li pokazuje typowy sposób projektowania algorytmu estymacji parametrów w większości algorytmów adaptacyjnych, np. w algorytmie adaptacyjnym Bayarda-Wena, algorytmie adaptacyjnym linearyzacji i adaptacyjnym algorytmie Sadegha-Horowitza.

### C.1 Algorytm Sadegha-Horowitza

Dowód stabilności algorytmu nieadaptacyjnego Sadegha-Horowitza przeprowadzimy metodą funkcji Lapunowa, [SH90]. W tym celu rozważymy model dynamiki sztywnego robota manipulacyjnego opisanego równaniem

$$\mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}) = \mathbf{u}. \quad (\text{C.1})$$

Zgodnie z algorytmem Sadegha-Horowitza, sterowanie podawane na wejście manipulatora ma postać

$$\mathbf{u} = \mathbf{Q}(\mathbf{q})\ddot{\mathbf{q}}_r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}_r + \mathbf{D}(\mathbf{q}) - \mathbf{K}_d\mathbf{s} - \mathbf{R}\mathbf{e}, \quad (\text{C.2})$$

gdzie  $\mathbf{e} = \mathbf{q} - \mathbf{q}_d$ ,  $\mathbf{s} = \dot{\mathbf{q}} - \dot{\mathbf{q}}_r = \dot{\mathbf{e}} + \boldsymbol{\Lambda}\mathbf{e}$ ,  $\mathbf{K}_d = \mathbf{K}_d^T > 0$ ,  $\mathbf{R} = \mathbf{R}^T > 0$ ,

$\Lambda = \Lambda^T > 0$ . Równania układu z zamkniętą pętlą sprzężenia zwrotnego przyjmują następującą postać

$$\begin{cases} \mathbf{Q}(\mathbf{q})\dot{\mathbf{s}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{s} + \mathbf{K}_d\mathbf{s} + \mathbf{R}\mathbf{e} = \mathbf{0} \\ \dot{\mathbf{e}} = -\Lambda\mathbf{e} + \mathbf{s} \\ \mathbf{q} = \mathbf{e} + \mathbf{q}_d. \end{cases} \quad (\text{C.3})$$

Jako funkcję Lapunowa wybierzemy

$$V(\mathbf{e}, \mathbf{s}, t) = \frac{1}{2}\mathbf{s}^T\mathbf{Q}(\mathbf{q})\mathbf{s} + \frac{1}{2}\mathbf{e}^T\mathbf{R}\mathbf{e}. \quad (\text{C.4})$$

Z dodatniej określoności macierzy bezwładności  $\mathbf{Q}(\mathbf{q})$  oraz definicji macierzy  $\mathbf{R}$  wynika, że  $V \geq 0$ . Pochodna  $\dot{V}$  wyliczona wzdłuż trajektorii układu (C.3) wynosi

$$\dot{V} = -\mathbf{s}^T\mathbf{C}\mathbf{s} - \mathbf{s}^T\mathbf{K}_d\mathbf{s} - \mathbf{s}^T\mathbf{R}\mathbf{e} + \frac{1}{2}\mathbf{s}^T\dot{\mathbf{Q}}\mathbf{s} + \mathbf{e}^T\dot{\mathbf{R}}\mathbf{e}.$$

Korzystając z własności (5.6) oraz z definicji zmiennej ślizgu  $\mathbf{s}$  otrzymujemy

$$\dot{V} = -\mathbf{s}^T\mathbf{K}_d\mathbf{s} - \mathbf{e}^T\mathbf{R}\Lambda\mathbf{e}. \quad (\text{C.5})$$

Z własności form kwadratowych wiadomo, że

$$\underline{\lambda}_{\mathbf{K}_d}\|\mathbf{s}\|^2 \leq \mathbf{s}^T\mathbf{K}_d\mathbf{s} \leq \bar{\lambda}_{\mathbf{K}_d}\|\mathbf{s}\|^2, \quad (\text{C.6})$$

skąd wynika nierówność

$$-\mathbf{s}^T\mathbf{K}_d\mathbf{s} \leq -\underline{\lambda}_{\mathbf{K}_d}\|\mathbf{s}\|^2.$$

Podobnie, dla drugiego składnika w (C.5) mamy oszacowanie

$$-\mathbf{e}^T\mathbf{R}\Lambda\mathbf{e} \leq -\underline{\lambda}_{\mathbf{R}\Lambda}\|\mathbf{e}\|^2.$$

Po wstawieniu obu oszacowań do (C.5) otrzymujemy

$$\begin{aligned} \dot{V} &= -\mathbf{s}^T\mathbf{K}_d\mathbf{s} - \mathbf{e}^T\mathbf{R}\Lambda\mathbf{e} \leq \\ &\leq -\underline{\lambda}_{\mathbf{K}_d}\|\mathbf{s}\|^2 - \underline{\lambda}_{\mathbf{R}\Lambda}\|\mathbf{e}\|^2 \leq -\alpha(\|\mathbf{s}\|^2 + \|\mathbf{e}\|^2), \end{aligned} \quad (\text{C.7})$$

gdzie  $\alpha = \min(\underline{\lambda}_{\mathbf{K}_d}, \underline{\lambda}_{\mathbf{R}\Lambda})$ . Kolejnym krokiem dowodu jest oszacowanie funkcji Lapunowa  $V$  od góry. Ponownie skorzystamy z własności (C.6), aby otrzymać

$$\begin{aligned} V &= \frac{1}{2}\mathbf{s}^T\mathbf{Q}(\mathbf{q})\mathbf{s} + \frac{1}{2}\mathbf{e}^T\mathbf{R}\mathbf{e} \leq \\ &\leq \frac{1}{2}(\bar{\lambda}_{\mathbf{Q}}\|\mathbf{s}\|^2 + \bar{\lambda}_{\mathbf{R}}\|\mathbf{e}\|^2) \leq \beta(\|\mathbf{s}\|^2 + \|\mathbf{e}\|^2), \end{aligned} \quad (\text{C.8})$$

gdzie  $\bar{\lambda}_{\mathbf{Q}} = \max_{\mathbf{q}} \bar{\lambda}_{\mathbf{Q}(\mathbf{q})}$  oraz  $\beta = \frac{1}{2} \max(\bar{\lambda}_{\mathbf{Q}}, \bar{\lambda}_{\mathbf{R}})$ . Ostatecznie uzyskujemy

parę oszacowań dla funkcji Lapunowa i jej pochodnej

$$\begin{aligned} V &\leq \beta (\|\mathbf{s}\|^2 + \|\mathbf{e}\|^2), \\ \dot{V} &\leq -\alpha (\|\mathbf{s}\|^2 + \|\mathbf{e}\|^2), \end{aligned}$$

z których wynika nierówność

$$\dot{V} \leq -\frac{\alpha}{\beta} V, \quad (C.9)$$

a zatem

$$V(t) \leq V(0) \exp\left(-\frac{\alpha}{\beta} t\right), \quad (C.10)$$

co oznacza, że funkcja Lapunowa  $V$  zanika eksponencjalnie do 0. Po ponownym skorzystaniu z własności form kwadratowych i wzoru (C.10) otrzymujemy oszacowanie funkcji Lapunowa od dołu

$$\frac{1}{2} \underline{\lambda}_{\mathbf{Q}} \|\mathbf{s}\|^2 \leq \frac{1}{2} \mathbf{s}^T \mathbf{Q}(\mathbf{q}) \mathbf{s} + \frac{1}{2} \mathbf{e}^T \mathbf{R} \mathbf{e} = V(t) \leq V(0) \exp\left(-\frac{\alpha}{\beta} t\right), \quad (C.11)$$

w którym  $\underline{\lambda}_{\mathbf{Q}} = \min_{\mathbf{q}} \underline{\lambda}_{\mathbf{Q}(\mathbf{q})}$ . Z nierówności (C.11) wynika, że prawdziwe jest następujące oszacowanie

$$0 \leq \|\mathbf{s}\| \leq \sqrt{\frac{2V(0)}{\underline{\lambda}_{\mathbf{Q}}}} \exp\left(-\frac{\alpha}{2\beta} t\right).$$

Podobnie otrzymujemy

$$0 \leq \|\mathbf{e}\| \leq \sqrt{\frac{2V(0)}{\underline{\lambda}_{\mathbf{R}}}} \exp\left(-\frac{\alpha}{2\beta} t\right),$$

tak więc eksponencjalna zbieżność błędu śledzenia położenia  $\mathbf{e}$  jest zapewniona.

## C.2 Adaptacyjny algorytm Slotine'a-Li

Dowód stabilności adaptacyjnego algorytmu Slotine'a-Li przeprowadzimy w oparciu o Zasadę Niezmienniczości La Salle'a\*. Rozważmy model dynamiki manipulatora opisany równaniem (6.25) z nieznanymi parametrami zastąpimy

$$\mathbf{Q}(\mathbf{q}, \boldsymbol{\theta}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\theta}) \dot{\mathbf{q}} + \mathbf{D}(\mathbf{q}, \boldsymbol{\theta}) = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \boldsymbol{\theta} = \mathbf{u}. \quad (C.12)$$

---

\*Zobacz dodatek B.2.

Na wejście manipulatora podajemy sterowanie (6.40)

$$\begin{aligned} \mathbf{u} &= \mathbf{Q}(\mathbf{q}, \hat{\boldsymbol{\theta}}) \ddot{\mathbf{q}}_r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \hat{\boldsymbol{\theta}}) \dot{\mathbf{q}}_r + \mathbf{D}(\mathbf{q}, \hat{\boldsymbol{\theta}}) - \mathbf{K}_d \mathbf{s} = \\ &= \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r) \hat{\boldsymbol{\theta}} - \mathbf{K}_d \mathbf{s}, \end{aligned} \quad (C.13)$$

gdzie  $\mathbf{e} = \mathbf{q} - \mathbf{q}_d$ ,  $\mathbf{s} = \dot{\mathbf{q}} - \dot{\mathbf{q}}_r = \dot{\mathbf{e}} + \boldsymbol{\Lambda} \mathbf{e}$ ,  $\mathbf{K}_d = \mathbf{K}_d^T > 0$ ,  $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}^T > 0$ . Estymaty  $\hat{\boldsymbol{\theta}}$  nieznaných parametrów zastępczych modelu są uzyskiwane z równania

$$\dot{\hat{\boldsymbol{\theta}}}(t) = -\boldsymbol{\Gamma} \mathbf{Y}^T(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r) \mathbf{s}, \quad (C.14)$$

przy czym macierz  $\boldsymbol{\Gamma}$  jest symetryczną i dodatnio określoną macierzą wzmocnień adaptacyjnych o wymiarze  $p \times p$ , gdzie  $p$  jest liczbą nieznaných parametrów zastępczych modelu.

Równania układu z zamkniętą pętlą sterowania są następujące

$$\begin{cases} \mathbf{Q}(\mathbf{q}) \dot{\mathbf{s}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \mathbf{s} + \mathbf{K}_d \mathbf{s} = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r) \tilde{\boldsymbol{\theta}} \\ \dot{\hat{\boldsymbol{\theta}}}(t) = -\boldsymbol{\Gamma} \mathbf{Y}^T(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r) \mathbf{s} \\ \tilde{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t) - \boldsymbol{\theta} \\ \dot{\mathbf{e}} = -\boldsymbol{\Lambda} \mathbf{e} + \mathbf{s} \\ \mathbf{q} = \mathbf{e} + \mathbf{q}_d. \end{cases} \quad (C.15)$$

Oznaczmy symbolem  $\mathbf{x} = (\mathbf{e}^T, \mathbf{s}^T, \tilde{\boldsymbol{\theta}}^T)^T$  trajektorię układu zamkniętego i wybierzmy następującą funkcję Lapunowa

$$V(\mathbf{x}, t) = \frac{1}{2} \mathbf{s}^T \mathbf{Q}(\mathbf{q}) \mathbf{s} + \frac{1}{2} \tilde{\boldsymbol{\theta}}^T \boldsymbol{\Gamma}^{-1} \tilde{\boldsymbol{\theta}}. \quad (C.16)$$

Policzymy pochodną funkcji Lapunowa wzdłuż trajektorii układu z zamkniętą pętlą sterowania

$$\dot{V} = \frac{1}{2} \mathbf{s}^T \dot{\mathbf{Q}} \mathbf{s} - \mathbf{s}^T \mathbf{C} \mathbf{s} - \mathbf{s}^T \mathbf{K}_d \mathbf{s} + \mathbf{s}^T \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r) \tilde{\boldsymbol{\theta}} + \tilde{\boldsymbol{\theta}}^T \boldsymbol{\Gamma}^{-1} \dot{\tilde{\boldsymbol{\theta}}}.$$

Korzystając z własności skośnej symetrii (5.6) oraz z definicji algorytmu estymacji nieznaných parametrów  $\tilde{\boldsymbol{\theta}}$ , otrzymujemy wyrażenie na pochodną funkcji Lapunowa  $\dot{V}$  liczoną wzdłuż trajektorii układu (C.15)

$$\dot{V} = -\mathbf{s}^T \mathbf{K}_d \mathbf{s} = -W(\mathbf{x}) \leq 0. \quad (C.17)$$

Z Zasady Niezmienniczości La Salle'a wynika, że trajektorie układu (C.15) dążą do maksymalnego zbioru niezmienniczego zawartego w  $X = \{\mathbf{x} | \mathbf{s} = \mathbf{0}\}$ .



Ponieważ  $\mathbf{s} = \dot{\mathbf{e}} + \mathbf{\Lambda e} \rightarrow \mathbf{0}$ , to przy wyborze dodatnio określonej macierzy  $\mathbf{\Lambda}$  błąd śledzenia trajektorii  $\mathbf{e}(t)$  zanika do zera eksponencjalnie. Na zakończenie należy podkreślić, że wybrana funkcja Lapunowa nie gwarantuje zbieżności estymat parametrów zastępczych do ich wartości rzeczywistych.

## Literatura

- [SH90] N. Sadegh i R. Horowitz, Stability and robustness analysis of a class of adaptive controllers for robotic manipulators. *Int. J. Robotics Research*, **9**(3):74–94, 1990.



# Indeks

Wytłuszczone numery stron oznaczają miejsca, gdzie dane pojęcie zostało wprowadzone (zdefiniowane), pozostałe — miejsca, gdzie było ono użyte.

- A**
- aksjomat pierścienia ..... **402**
  - aksjomaty normy ..... **402**
  - algebra Liego ..... **88, 277**
    - grupy  $\mathbb{SE}(3)$  ..... **32**
    - pól wektorowych ..... **406**
    - układu ..... **274**
  - algorytm
    - $\lambda$ -śledzenia
      - manipulatora ..... **236**
      - robota mobilnego ..... **389**
    - adaptacyjny ..... **223**
      - Bayarda-Wena ..... **226**
      - linearyzacji ..... **224**
      - Sadegha-Horowitza ..... **229**
      - Slotine'a-Li ..... **228, 419**
    - całkowania wstecznego ..... **255, 261**
    - Corona-Pometa ..... **370, 393**
    - Denavita-Hartenberga **42, 44, 46, 48, 50, 64, 89, 356**
    - dysypatywny ..... **209**
    - estymacji ..... **224, 226, 230, 417**
    - jakobianu
      - dołączonego ..... **163, 175**
      - odpornego ..... **170**
      - odwrotnego ..... **103**
      - pseudoodwrotnego ..... **111**
      - pseudoodwrotnego z projekcją ... **117**
      - rozszerzonego ..... **119**
      - transponowanego ..... **122**
    - kinematyki
      - prostych kołowych robotów mobilnych ..... **360, 392**
      - złożonych kołowych robotów mobilnych ..... **366, 392**
      - kinematyki odwrotnej
        - asymptotyczny ..... **104, 112, 117, 121**
        - manipulatora ..... **95**
        - osobliwy ..... **140**
        - robotu mobilnego ..... **182, 183**
      - linearyzacji
        - dynamicznej ..... **373**
        - statycznej ..... **213, 251**
      - mnożników Lagrange'a ..... **125**
      - najszybszego spadku ..... **122**
      - Newtona ..... **104, 187**
        - z projekcją ..... **117**
      - Newtona-Smale'a ..... **163, 175**
      - obliczanego momentu ..... **209**
      - odsprężania ..... **204**
      - Ortegi-Lorii ..... **259**
      - planowania Lie-algebraicznego ..... **305**
      - przestrzeni zerowej ..... **166**
      - Qu-Dorsey'a ..... **236**
      - Sadegha-Horowitza ..... **218, 417**
      - Slotine'a-Li ..... **216**
        - sterowania robotu mobilnego ..... **386**
      - stabilizacji ..... **372**
      - sterowania ..... **203, 209, 224, 369**
        - ślizgowego ..... **232**
        - uniwersalny ..... **219**
      - Walsha-Tilbury'ego-Sastry'ego-Muraya-Laumonda ..... **377**
      - Wena-Bayarda ..... **214, 216**
    - aspekt kinematyki ..... **57**
    - atan2 ..... **99**
    - atlas rozmaitości ..... **31**

**B**

    - baza Ph. Halla ..... **277, 281, 297, 300**

- brzytwa Ockhama ..... 18
- C**
- całka pierwsza ..... 138, 406
- ciało sztywne ..... 27, 40, 88
- czas fizyczny ..... 25
- D**
- delta Kroneckera ..... 200
- długość wektora ..... 25
- dowód stabilności ..... 417, 419
- dyfeomorfizm ..... 133, 252, 404, 405
- dynamika
- manipulatora
    - o elastycznych przegubach ... 249
    - sztywnego ..... 201
  - robota mobilnego ..... 367
- dystrybucja ..... 37, 88
- dywergencja ..... 132, 136
- działanie pola wektorowego ..... 405
- E**
- efektor manipulatora ..... 39, 41
- eksperymenty ..... 175, 241
- ekstremala ..... 286, 408
- osobliwa ..... 87, 185, 408
  - regularna ..... 320
- ekwiwariantność ..... 28
- elementarne
- obroty ..... 43
  - przesunięcia ..... 43
- elipsoida
- manipulowalności ..... 128, 174
  - podwójnego wahadła ..... 130
  - mobilności ..... 186, 188
- energia
- kinetyczna ..... 194, 195, 247
  - manipulatora ..... 200
  - robota mobilnego ..... 367
  - potencjalna ..... 194, 200, 249
  - manipulatora ..... 201
- euklidesowa
- norma ..... 25
  - przestrzeń ..... 25
- F**
- filtracja dystrybucji ..... 38
- forma kwadratowa ..... 111, 402
- formuła
- Campbella-Bakera-Hausdorffa-Dyn-  
kina ..... 278, 281, 298, 310
  - uogólniona ..... 299, 307, 310
- CBHD ..... 278
- funkcja
- analityczna ..... 403
  - gładka ..... 403
  - Hamiltona ..... 319, 406
  - Lapunowa ..... 214, 217, 220, 225, 227,  
230, 233, 237, 387, 411, 417, 420
  - Nussbauma ..... 212
  - ograniczona ..... 403
  - przełączająca ..... 319
  - sterująca ..... 406
- G**
- generatory układu ..... 273, 293
- gładka rozmaitość ..... 30
- grupa Liego ..... 31, 88
- H**
- halsowanie ..... 274
- hamiltonian ..... 86, 134, 286, 406, 408
- hiperboliczna postać normalna kinematy-  
ki manipulatora ..... 146, 148
- homeomorfizm ..... 403
- I**
- iloczyn wektorów
- mieszany ..... 27
  - skalaryny ..... 25
  - wektorowy ..... 25
- indeks konfiguracji osobliwej ..... 138
- inwariantność ..... 28
- J**
- jakobian
- analityczny
  - manipulatora 61, 71, 84, 111, 117,  
128, 133, 163, 204
  - manipulatora IRb-6 ..... 63, 75
  - manipulatora SCARA ..... 63, 74
  - podwójnego wahadła ..... 62, 73
  - potrójnego wahadła ..... 62
  - roboty mobilnego ..... 86, 90, 290
  - rozszerzony ..... 119, 174
- geometryczny

- manipulatora ..... **63, 71, 79, 89**  
 manipulatora IRb-6 ..... **67, 75**  
 manipulatora *SCARA* ..... **66, 71, 74, 77**  
 podwójnego wahadła ..... **65, 70, 73, 77**  
 potrójnego wahadła ..... **66**  
 rozszerzony ..... **125**  
 w ciele ..... **63, 69**  
 w przestrzeni ..... **63, 65, 125**  
 manipulatora ..... **68, 89**  
 manipulatora *SCARA* ..... **69**  
 podwójne wahadło ..... **68**  
 potrójne wahadło ..... **69**
- K**
- kąty  
 Eulera ..... **34, 60, 71**  
 kołysanie-kiwanie-myszkowanie... **35**
- kinematyka  
 manipulatora ..... **41, 43, 95**  
 IRb-6 ..... **49, 60**  
 korzędu 1 ..... **144**  
 osobliwa ..... **144**  
 podwójne wahadło ..... **44, 53, 56**  
 potrójne wahadło ..... **46, 57**  
*SCARA* ..... **47, 54, 59**  
 w aspekcie ..... **57, 99, 205, 339**  
 we współrzędnych ..... **55, 131**  
 robota mobilnego ..... **83, 84, 90, 181**  
 układu robotycznego ..... **39**
- konfiguracja  
 izotropowa manipulatora ..... **129**  
 osobliwa  
 manipulatora ..... **79, 96, 134**  
 manipulatora IRb-6 .. **82, 139, 158**  
 manipulatora *SCARA* ..... **82**  
 możliwa do uniknięcia .. **131, 135, 138, 174**  
 niemożliwa do uniknięcia .... **132, 135, 138, 174**  
 podwójnego wahadła ..... **80, 148**  
 potrójnego wahadła ..... **81**  
 regularna manipulatora ..... **79, 96**
- korząd osobliwości ..... **79, 136**
- kowymiar  
 rozmaiłości ..... **30**  
 zbioru osobliwości ..... **80**
- krzywizna toru ..... **26**  
 kwadratowa postać normalna kinematyki  
 manipulatora .... **145, 146, 147**
- L**
- lagranżian ..... **86, 194**  
 $\text{\LaTeX}$  ..... **21**  
 lemat  
 Barbalata ..... **258, 413**  
 Wena-Bayarda ..... **240, 413**
- linearyzacja ..... **203, 209, 213, 253**  
 dynamiczna ..... **370, 374, 393**  
 statyczna ..... **369, 393**
- M**
- macierz  
 bezwładności ..... **209**  
 dołączona ..... **164, 401**  
 dodatnio określona ..... **195**  
 górno-trójkątna ..... **249**  
 Grama ..... **183**  
 Hessego ..... **138**  
 inercji ..... **195, 196, 199**  
 Jacobiego ..... **61, 403**  
 manipulowalności ..... **128**  
 mobilności ..... **186**  
 obrotu ..... **28**  
 odwrotna ..... **111, 170**  
 odwrotność uogólniona ..... **402**  
 ortogonalna ..... **28**  
 przekształcenia liniowego ..... **401**  
 przybliżenia liniowego ..... **133, 136**  
 pseudoodwrotność ..... **402**  
 regresji ..... **223**  
 skośnie symetryczna ..... **31**  
 sterowań ..... **353**  
 symetryczna ..... **195**  
 zwrrotna odwrotność uogólniona. **402**
- manipulator  
*EDDA* ..... **228**  
 algorytm  $\lambda$ -śledzenia ..... **238**  
 algorytm adaptacyjny Bayarda-  
 -Wena ..... **228**  
 algorytm adaptacyjny linearyzacji  
**225**  
 algorytm adaptacyjny Sadegha-Ho-  
 rowitza ..... **231**

- algorytm adaptacyjny Slotine'a-Li  
     229  
 algorytm linearyzacji statycznej ..  
     214  
 algorytm odsprężania ..... 205  
 algorytm Qu-Dorsey'a ..... 236  
 algorytm Sadegha-Horowitz'a .. 219  
 algorytm Slotine'a-Li ..... 218  
 algorytm sterowania ślizgowego ..  
     235  
 algorytm uniwersalny ..... 221  
 algorytm Wena-Bayarda ..... 216  
 dynamika ..... 202, 223  
 kinematyka ..... 57  
 parametry dynamiki ..... 202
- elastyczny *EDDA*  
 algorytm całkowania wstecznego ..  
     258  
 algorytm linearyzacji statycznej ..  
     254  
 algorytm Ortegi-Lorii ..... 260  
 dynamika ..... 250
- IRb-6  
 jakobian analityczny ..... 63, 75  
 jakobian geometryczny ..... 67, 75  
 kinematyka ..... 49, 60  
 konfiguracje osobliwe ..... 82  
 osobliwe zadanie odwrotne ..... 161  
 parametry Denavita-Hartenberga ..  
     49  
 postacie normalne kinematyki 158  
 przestrzeń robocza ..... 61  
 unikanie osobliwości ..... 139  
 zadanie odwrotne .. 100, 109, 114,  
     118, 121, 123, 172  
 nieredundantny ..... 55  
 o elastycznych przegubach . 246, 251,  
     254
- podwójne wahadło  
 elipsoidalności ..... 130  
 jakobian analityczny ..... 62, 73  
 jakobian geometryczny . 65, 70, 73,  
     77  
 jakobian manipulatora ..... 68  
 kinematyka ..... 44, 53, 56  
 konfiguracje osobliwe ..... 80  
 osobliwe zadanie odwrotne... 151,
- 156  
 parametry Denavita-Hartenberga ..  
     45  
 postacie normalne kinematyki 148  
 zadanie odwrotne ..... 98, 102, 104,  
     165, 167, 170
- potrójne wahadło  
 jakobian analityczny ..... 62  
 jakobian geometryczny ..... 66  
 jakobian manipulatora ..... 69  
 kinematyka ..... 46, 57  
 konfiguracje osobliwe ..... 81  
 parametry Denavita-Hartenberga ..  
     46  
 zadanie odwrotne ..... 112, 126  
 redundantny ..... 55, 111, 125, 131  
*SCARA* ..... 44  
 jakobian analityczny ..... 63, 74  
 jakobian geometryczny . 66, 71, 74,  
     77  
 jakobian manipulatora ..... 69  
 kinematyka ..... 47, 54, 59  
 konfiguracje osobliwe ..... 82  
 parametry Denavita-Hartenberga ..  
     48  
 przestrzeń robocza ..... 60  
 zadanie odwrotne ..... 103  
 sztywny ..... 40, 196, 213  
 manipulowalność . 118, 127, 129, 174, 187  
 mapa rozmaiwości ..... 30  
 zgodność ..... 31
- mechanika  
 hamiltonowska ..... 206  
 lagranżowska ..... 206  
 wakonomiczna ..... 392
- metoda  
 algebraiczna ..... 98  
 geometryczna ..... 101  
 jakobianu  
 dołączonego ..... 163, 175  
 odpornego ..... 168  
 odwrotnego ..... 103  
 pseudoodwrotnego ..... 111  
 pseudoodwrotnego z projekcją ...  
     117  
 rozszerzonego ..... 119  
 transponowanego ..... 122

- Lapunowa II ..... 412, 417  
 Laumonda ..... 347  
 Lie-algebraiczna ..... 285, 297, 310  
 mnożników Lagrange'a ..... 125  
 najmniejszych kwadratów ..... 168  
 Newtona ..... 182, 285, 287  
 odsprzęgania ..... 206  
 optymalnego planowania  
   ogólna ..... 285  
   specjalizowana ..... 317  
 podcelów ..... 317, 341, 348  
 postaci normalnych ... 140, 142, 175  
 propagacji fali ..... 310  
 przestrzeni zerowej ..... 166, 175  
 sterowań sinusoidalnych ... 317, 331, 348  
 Sussmanna-Lafferiera ..... 348  
 uśredniania ..... 285, 292, 310
- metody  
 dyskretne ..... 310  
 jacobianowe ..... 103  
 potencjałowe ..... 311  
 przeszukiwania grafu ..... 311
- mnożenie  
 skalarne ..... 25  
 wektorowe ..... 25
- mnożniki Lagrange'a ..... 126, 352, 353  
 mobilność ..... 186, 187
- model dynamiki  
 manipulatora o elastycznych przegubach ..... 246  
 manipulatora sztywnego ... 209, 210, 213
- N
- nawias Liego ..... 38, 87, 88, 274, 405  
 iterowany ..... 406  
 własności ..... 276
- nieznajomość  
 modelu ..... 210  
   parametryczna ..... 210, 222, 370  
   strukturalna ..... 210, 231, 370
- norma  
 aksjomaty ..... 402  
 euklidesowa ..... 25, 201, 402  
 macierzy ..... 402  
   indukowana ..... 402
- widmowa ..... 402
- O
- obroty elementarne ..... 43  
 odsprzężenie ..... 203  
 odwrotność  
   przybliżona macierzy ..... 168  
   uogólniona macierzy ..... 402  
   zwrotna ..... 402
- odwzorowanie  
 analityczne ..... 403  
 gładkie ..... 403  
 klasy  $\mathbb{B}$  ..... 403  
 klasy  $C^k$  ..... 403  
 klasy  $L_k$  ..... 403  
 lokalnie lipschitzowskie ... 403, 414  
 mierzalne ..... 403  
 odwrotne ..... 403, 404  
 ograniczone ..... 403
- ogniwo manipulatora ..... 40
- ograniczenia  
 fazowe ... 36, 38, 194, 352, 357, 379  
   postać Pfaffa ..... 36  
 holonomiczne ..... 37, 38, 89, 265  
 konfiguracyjne ..... 36, 38, 194, 362  
 mechaniczne przegubów ..... 118  
 nieholonomiczne .. 37, 38, 83, 88, 89, 265  
   brak poślizgu kół ..... 266  
   kryterium ..... 38  
   zasada zachowania momentu pędu ..... 267
- osobliwości  
 manipulatora .. 79, 90, 129, 131, 167, 173  
 reprezentacji ..... 80  
 robota mobilnego ..... 86
- osobliwy algorytm kinematyki odwrotnej ..... 140
- oś obrotu ..... 34
- P
- para kinematyczna ..... 40  
 paraboliczna postać normalna kinematyki manipulatora ..... 146, 147
- parametry  
 barycentryczne ..... 381, 383, 386  
 Denavita-Hartenberga ..... 43

- manipulatora IRb-6 ..... 49
- manipulatora *SCARA* ..... 48
- podwójnego wahadła ..... 45
- potrójnego wahadła ..... 46
- dynamiki ..... 222
  - manipulatora *EDDA* ..... 202
  - geometryczne ..... 43, 365, 381
  - zastępcze ..... 419
- parametryzacja ..... 30
  - Cayleya ..... 34, 73
  - kąty Eulera ..... 34
  - kołysanie-kiwanie-myszkowanie .. 35, 72
  - wykładnicza ..... 51
    - $SE(3)$  ..... 35
- planowanie ruchu ..... 285, 317
  - metody
    - globalne ..... 271
    - lokalne ..... 272
    - ogólnego przeznaczenia ..... 272
    - specjalizowane ..... 272
- pola
  - hamiltonowskie ..... 136, 174
  - sterujące ..... 83
  - stowarzyszone ..... 273
- pole wektorowe ..... 37, 404
  - ruchu własnego ..... 166
- polisystem dynamiczny ..... 407
- postać normalna ..... 141, 174
  - dynamiki ..... 369
  - kinematyki manipulatora ..... 141
    - hiperboliczna ..... 146, 147, 148, 175
    - IRb-6 ..... 158
    - kwadratowa .. 145, 146, 147, 175
    - paraboliczna ..... 146, 147, 175
    - podwójne wahadło ..... 148
    - wokół konfiguracji osobliwej .. 144
    - wokół konfiguracji regularnej .. 141
- postać prenormalna ..... 153
- powtarzalność ..... 174
- prędkość
  - ciała sztywnego ..... 31
    - w układzie ciała ..... 31
    - w układzie przestrzeni ..... 31, 125
  - kątowa
    - w układzie ciała ..... 31
    - w układzie przestrzeni ..... 31
- liniowa
  - w układzie ciała ..... 32
  - w układzie przestrzeni ..... 32
- przegubów ..... 125
- punktu materialnego ..... 26
- prędkości
  - przegubowe ..... 128
  - uogólnione ..... 36, 40
  - zadaniowe ..... 128
- przełożenie przekładni ..... 199
- przegub manipulatora
  - obrotowy ..... 40
  - przesuwny ..... 40
- przemieszczenie ciała sztywnego ..... 28
- przestrzeń
  - euklidesowa ..... 25
  - fizyczna ..... 25
  - Hilberta ..... 83, 403
  - konfiguracyjna
    - ciała sztywnego ..... 29
    - manipulatora ..... 55
    - punktu materialnego ..... 26
  - przegubowa manipulatora .. 55, 126, 135, 206
- robocza ..... 96
  - manipulatora IRb-6 ..... 61
  - manipulatora *SCARA* ..... 60
- stanu ..... 84, 406
- sterowań ..... 84
- styczna ..... 30, 32, 36, 38, 79
- zadaniowa manipulatora 55, 96, 122, 164, 166, 194, 206
- przesunięcia elementarne ..... 43
- przyczepa ..... 351
- przyspieszenie ruchu ..... 26
- pseudoodwrotność
  - jakobianu analitycznego
    - manipulatora ..... 112
    - robota mobilnego ..... 183
  - macierzy ..... 111, 170, 294, 402
- punkt
  - osobliwy ..... 405
  - równowagi .. 132, 134, 136, 405, 407
    - asymptotycznie stabilny ..... 133
    - eksponencjalnie stabilny ..... 412
    - globalnie eksponencjalnie stabilny 413



- globalnie jednostajnie asymptotycznie stabilny ..... **412, 413**  
 globalnie jednostajnie stabilny ... **412**  
 jednostajnie asymptotycznie stabilny ..... **412**  
 jednostajnie stabilny ..... **412**  
 jednostajnie stabilny asymptotycznie globalnie ..... **371**  
 niestabilny ..... **133**  
 stabilny w sensie Lapunowa... **133**  
 regularny ..... **405**
- R**
- ramię manipulatora ..... **40**  
 regulator PD ..... **203, 213, 218–220, 235**  
 reprezentacja kinematyki  
 kwaternionowa ..... **89**  
 we współrzędnych ..... **55**  
 wg Denavita-Hartenberga ..... **43**  
 wykładnicza ..... **52, 76, 89**  
 RL-równoważność ..... **141**  
 lokalna ..... **141**  
 robot mobilny ..... **83**  
 jednokołowy ..... **275, 317, 357**  
 metoda Lie-algebraiczna ..... **305**  
 metoda uśredniania ..... **295**  
 kołowy ..... **351**  
 charakterystyka koła ..... **355**  
 klasyfikacja ..... **361, 392**  
 prosty ..... **351, 358**  
 złożony ..... **351, 366**  
 Ulisses ..... **365**  
 $\lambda$ -śledzenie ..... **390**  
 algorytm Corona-Pometa ..... **372**  
 algorytm linearyzacji dynamicznej **374**  
 algorytm Walsha-Tilbury'ego-Sastri'ego-Murraya-Laumonda **378**  
 model dynamiki ..... **368**  
 model kinematyki ..... **365**  
 postać normalna ..... **372**  
 sterowanie adaptacyjne ..... **388**  
 współrzędne linearyzujące... **382, 385**  
 robot nieholonomiczny ..... **83**  
 rozmaitość  
 gładka ..... **30**  
 konfiguracyjna .. **36, 38–41, 131, 265**  
 przegubowa ..... **41, 173**  
 różniczkowa ..... **88**  
 ruchu własnego ..... **132, 174**  
 zadaniowa .. **39, 41, 79, 95, 124, 131, 174**  
 różniczka Gâteaux ..... **84**  
 równania  
 dynamiki ..... **195**  
 bez ograniczeń ..... **196**  
 manipulatora o elastycznych przegubach ..... **246, 249**  
 manipulatora sztywnego ..... **201**  
 układu robotycznego ..... **195**  
 z ograniczeniami ..... **353**  
 Eulera-Lagrange'a ..... **195, 249, 409**  
 kanoniczne Hamiltona. **286, 318, 408**  
 kinematyki ..... **39**  
 równanie  
 Lapunowa ..... **225**  
 wariacyjne ..... **85**  
 równoważność ..... **140, 141**  
 równozmienność ..... **28**  
 ruch punktu materialnego ..... **26**
- S**
- scenariusz ruchu ..... **322**  
 sfera nieholonomiczna ..... **311**  
 siły  
 Coriolisa ..... **195**  
 grawitacji ..... **195**  
 odśrodkowe ..... **195**  
 sprężystości ..... **245**  
 tarcia ..... **196**  
 uogólnione ..... **195**  
 skręcenie toru ..... **27**  
 skrętnik ..... **52, 77, 89**  
 spadający kot ..... **267**  
 specjalna grupa  
 euklidesowa ..... **29, 41, 63, 88**  
 obrotów ..... **28, 88**  
 sprzężenie zwrotne  
 dynamiczne ..... **212, 250, 261**  
 statyczne... **203, 213, 253, 261, 369, 370, 392**  
 stabilizowalność ..... **369, 407**

stabilność  
   asymptotyczna ..... 224, 234  
   globalna ..... 225, 230  
   półglobalna ..... 216, 221, 228, 260  
   eksponencjalna ..... 213, 216–218, 229,  
     253, 378, 384  
 stała ruchu ..... 406  
 sterowalność ..... 273, 369, 407  
   lokalna ..... 407  
   w krótkim czasie ..... 273, 407  
   w krótkim czasie ..... 84, 407  
 sterowanie ..... 406  
   adaptacyjne ..... 210  
   nieosobliwe ..... 183, 185  
   odcinkami stałe ..... 288  
   odporne ..... 210  
   optymalne ..... 86  
   osobliwe ..... 86, 87, 186  
   sinusoidalne ..... 303  
 stopień  
   anizotropii ..... 129  
   mobilności ..... 361, 367  
   nawiasu Liego ..... 276  
   redundancji ..... 132, 134, 136  
   sterowalności ..... 361  
   swobody ..... 36  
   uwarunkowania ..... 130  
 stratyfikacja Whitneya ..... 80  
 strumień pola wektorowego ..... 278, 404  
 suriekcja liniowa ..... 79  
 symbole Christoffela I rodzaju ..... 195

## Ś

ścieżka ruchu ..... 26  
 ślad macierzy ..... 34  
 śledzenie trajektorii ..... 213

## T

toczenie się koła  
   bez poślizgu ..... 355  
   bocznego ..... 356  
   wzdłużnego ..... 356  
 tor ruchu ..... 26  
 torsja toru ..... 27  
 tożsamość Jacobiego ..... 276, 277, 334  
 trajektoria  
   optymalna ..... 325  
   przegubowa ..... 104, 166, 194, 214

  ruchu ..... 26  
   zadaniowa ..... 95, 194  
 traktor ..... 351, 358  
 trójkołowy ..... 359, 364  
   model dynamiki ..... 367  
   model kinematyki ..... 364  
   ograniczenia fazowe ..... 359  
 trójścian Freneta ..... 26  
 twierdzenie  
   Brocketta ..... 392  
   Chow ..... 89, 187, 275, 297  
   Frobeniusa ..... 89  
   La Salle'a-Yoshizawy ..... 234, 241, 258,  
     388, 413  
   o funkcji odwrotnej ..... 133, 403  
   o postaci normalnej ..... 145, 146  
   o prostowaniu ..... 405  
   o uśrednianiu ..... 292  
   o wartości średniej ..... 220, 404  
   Stokesa ..... 317

## U

układ  
   2-łańcuchowy ..... 335  
   2-wejściowy generowanie kierunków  
     301  
   Brocketta ..... 275, 280, 331  
   estymacji ..... 211  
   łańcuchowy ..... 333, 348  
   napędowy ..... 196, 246  
   nieholonomiczny ..... 39, 293, 393  
   nilpotentny ..... 277  
   robotyczny ..... 36, 83, 352  
   uniwersalny ..... 394  
   współrzędnych ..... 30  
 układ dynamiczny ..... 404  
   autonomiczny ..... 405  
   hamiltonowski ..... 134, 174, 406, 408  
   nieautonomiczny ..... 104, 405, 412–414  
   ruchu własnego ..... 132, 134, 166  
   strumień ..... 404  
 układ sterowania ..... 39, 369, 406, 408  
   afiniczny ..... 203, 377  
   autonomiczny ..... 406  
   liniowy ..... 406  
   nieautonomiczny ..... 406  
   niestacjonarny ..... 406

niezależny od czasu.....	<b>406</b>
afiniczny.....	<b>407</b>
bezdryfowy.....	<b>83, 407</b>
liniowy.....	<b>407</b>
stacjonarny.....	<b>406</b>
uniwersalny.....	<b>211, 239</b>
zależny od czasu.....	<b>406</b>
liniowy.....	<b>406</b>

układy hamiltonowskie..... **136**

uniwersum

  fazowe..... **36, 39, 83**

  konfiguracyjne..... **36, 89**

## W

warstwa pól wektorowych..... **276**

warunek

  amplitudowy..... **304**

  częstotliwościowy..... **304**

  fazowy..... **304**

  Lipschitza..... **403**

  rzędu algebry Liego..... **275**

  transwersalności..... **163, 167**

warunki Moore'a-Penrose'a..... **402**

wektor wzrostu dystrybucji..... **38**

wiązka styczna..... **30**

własność osiągalności..... **369**

współrzędne

  jednorodne..... **28, 196**

  kartezjańskie..... **33, 71**

  konfiguracyjne..... **55**

  linearyzujące..... **379, 384**

  oś-kąt..... **33**

  przegubowe..... **55, 61, 133**

  sferyczne..... **33**

  uogólnione..... **36, 246**

  walcowe..... **33**

  wykładnicze w  $SE(3)$ ..... **35**

  wykładnicze w  $SO(3)$ ..... **34**

  wykładnicze w  $SE(3)$ ..... **88**

  zadaniowe..... **55, 61, 126**

wyjścia linearyzujące..... **374, 382**

wymiar rozmaitości..... **30**

## Z

zadanie

$\lambda$ -śledzenia..... **389**

  czaso-optymalne..... **319**

  Dubinsa..... **318, 347**

izoperymetryczne..... **182**

odwrotne..... **140**

  ciągłe..... **95**

  dla manipulatora IRb-6..... **100, 109,**

**114, 118, 121, 123, 161, 172**

  dla manipulatora SCARA..... **103**

  dla podwójnego wahadła..... **98, 102,**

**104, 151, 156, 165, 167, 170**

  dla potrójnego wahadła..... **112, 126**

  kinematyki manipulatora..... **95, 117,**

**121, 126, 164**

  kinematyki robota mobilnego..... **182**

  osobliwe..... **97, 131**

  punktowe..... **96, 122, 125**

  regularne..... **97**

  optymalizacji..... **112, 126**

  planowania ruchu..... **269**

    holonomiczne..... **269**

    nieholonomiczne..... **269**

    robota mobilnego..... **182**

  proste kinematyki manipulatora..... **95**

  Reedsa-Sheppa..... **318, 347**

  śledzenia trajektorii..... **202**

  sterowania..... **408**

  adaptacyjnego robota mobilnego.....

**386**

  optymalnego..... **86**

  robota mobilnego..... **384**

Zasada

  d'Alemberta..... **352, 380, 392**

  Maksimum Pontriagina..... **86, 90, 286,**

**332, 408**

  Najmniejszego Działania..... **195, 392**

  Niezmienności La Salle'a..... **411, 419**

zbiór

  I kategorii..... **80**

  analityczny..... **80**

  niezmienniczy..... **137, 405, 411, 420**

zmienna dołączona..... **87, 286, 408**

zręczność manipulatora..... **79, 129, 174**



# Problemy Współczesnej Nauki

## Teoria i Zastosowania

---

### Informatyka

Wykaz książek, które ukazały się w tej serii:

1. A. Mykowiecka, *Podstawy przetwarzania języka naturalnego. Metody generowania tekstów*
2. E. Dobryjanowicz, *Podstawy przetwarzania języka naturalnego. Wybrane metody analizy składniowej*
3. L. Bolc, A. Mykowiecka, *Podstawy przetwarzania języka naturalnego. Wybrane metody formalnego zapisu składni*
4. L. Bolc, J. Zaremba, *Wprowadzenie do uczenia się maszyn*
5. A. Szałas, *Zarys dedukcyjnych metod automatycznego wnioskowania*
6. R. Tadeusiewicz, *Sieci neuronowe*
7. W. Mokrzycki, *Encyklopedia przetwarzania obrazów*
8. M. Muraszkiewicz, H. Rybiński, *Bazy danych*
9. W. Skarbek, *Metody reprezentacji obrazów cyfrowych*
10. E. Gatnar, *Metody modelowania jakościowego*
11. M. Byrska, *Ochrona programu komputerowego w nowym prawie autorskim*
12. K. Subieta, *Ingres — system zarządzania relacyjną bazą danych*
13. J. Korbicz, A. Obuchowicz, D. Uciński, *Sztuczne sieci neuronowe — podstawy i zastosowania*
14. Z. Suraj, B. Komarek, *GRAF — Graficzny system konstruowania i symulowania sieci Petriego*
15. J. Chromiec, E. Strzemieczna, *Sztuczna inteligencja. Metody konstrukcji i analizy systemów eksperckich*
16. J. Cytowski, *Algorytmy genetyczne*
17. A. Wakulicz-Deja, *Podstawy systemów wyszukiwania informacji. Analiza metod*
18. L. Polkowski, M. Szczuka, *Elementy matematyki dla studentów kierunków informatycznych*
19. L. Bolc, K. Dziewicki, P. Rychlik, A. Szałas, *Wnioskowanie w logikach nieklasycznych. Podstawy teoretyczne*
20. Z. Vetulani, W. Abramowicz, G. Vetulani, *Język i technologia*
21. G. Bryll, *Metody odrzucania wyrażeń*
22. E. Rafajłowicz, W. Myszka, *LATEX — zaawansowane narzędzia*
23. D. Figura, *Obiektowe bazy danych*
24. E. Hajnicz, *Reprezentacja logiczna wiedzy zmieniającej się w czasie*
25. D. Rutkowska, *Inteligentne systemy obliczeniowe. Algorytmy genetyczne i sieci neuronowe w systemach rozmytych*
26. L. Banachowski, *Bazy danych. Tworzenie aplikacji*
27. W. Skarbek (Red.), *Multimedia. Algorytmy i standardy kompresji*
28. P. Zieliński, *Systemy informatyczne dla bankowości w zarysie*
29. P. Adamczewski, *Wdrożeniowe uwarunkowania zintegrowanych systemów informatycznych*
30. K. Wojciechowski, *System z niepewnością ograniczoną. Model, analiza i modyfikacja. Podejmowanie decyzji i reguły decyzyjne*
31. A. Czyżewski, *Dźwięk cyfrowy. Wybrane zagadnienia teoretyczne, technologia, zastosowania*

32. K. Subieta, *Obiektywność w projektowaniu i bazach danych*
33. L. Bolc, K. Dziewicki, P. Rychlik, A. Szałas, *Wnioskowanie w logikach nieklasycznych. Automatyzacja wnioskowania*
34. K. Subieta, *Słownik terminów z zakresu obiektywności*
35. M. Nieniewski, *Morfologia matematyczna w przetwarzaniu obrazów*
36. E. Lipowska-Nadolska, M. Morawski, *Tablice systoliczne. Problemy wybrane*
37. R. Tadeusiewicz, *Elementarne wprowadzenie do techniki sieci neuronowych z przykładowymi programami*
38. A. Mrózek, L. Płonka, *Analiza danych metodą zbiorów przybliżonych. Zastosowania w ekonomii, medycynie i sterowaniu*
39. A. Piegat, *Modelowanie i sterowanie rozmyte*
40. A. Janiak, *Wybrane problemy i algorytmy szeregowania zadań i rozdziału zasobów*
41. W. Skarbek, *Multimedia. Sprzęt i oprogramowanie*
42. J. Cytowski, *Metody i algorytmy sztucznej inteligencji w cyfrowym przetwarzaniu sygnałów*

## Statystyka

Wykaz książek, które ukazały się w tej serii:

1. J. R. Thompson, J. Koronacki, *Statystyczne sterowanie procesem. Metoda Deminga etapowej optymalizacji jakości*
2. E. Rafajłowicz, *Algorytmy planowania eksperymentu z implementacjami w środowisku MATHEMATICA*

## Zarządzanie

Wykaz książek, które ukazały się w tej serii:

1. R. Janikowski, *Zarządzanie ekologiczne*

## W przygotowaniu znajdują się następujące pozycje:

1. J. Doroszewski, *Struktura wiedzy i rozwiązywania problemów medycznych*
2. M. A. Kłopotek, *Konstrukcja systemów bazodanowych w światowej sieci komputerowej WWW*
3. J. Kulikowski, *Komputerowe przetwarzanie obrazów biomedycznych*
4. H. Kwaśnicka, *Obliczenia ewolucyjne w sztucznej inteligencji*
5. J. Mańdziuk, *Sieci neuronowe typu Hopfielda. Teoria i przykłady zastosowań*
6. R. Pruchnik, *Algorytmy odwracalnej kompresji danych*
7. A. Skowron, *Podstawy sztucznej inteligencji*
8. J. Stokłosa, T. Bilski, T. Pankowski, *Bezpieczeństwo danych w systemach informatycznych*
9. J. Wywiół, *Elementy metody reprezentacyjnej z wykorzystaniem statystycznego pakietu SPSS*