



Modeling, Performance Analysis and Control of Robot Manipulators

Edited by
Etienne Dombre
Wisama Khalil

ISTE

This page intentionally left blank

Modeling, Performance Analysis and Control of Robot Manipulators

This page intentionally left blank

Modeling, Performance Analysis and Control of Robot Manipulators

Edited by
Etienne Dombre
Wisama Khalil

ISTE

Part of this book adapted from “Analyse et modélisation des robots manipulateurs” and “Commande des robots manipulateurs” published in France in 2001 and 2002 by Hermès Science/Lavoisier

First published with revisions in Great Britain and the United States in 2007 by ISTE Ltd

Apart from any fair dealing for the purposes of research or private study, or criticism or review, as permitted under the Copyright, Designs and Patents Act 1988, this publication may only be reproduced, stored or transmitted, in any form or by any means, with the prior permission in writing of the publishers, or in the case of reprographic reproduction in accordance with the terms and licenses issued by the CLA. Enquiries concerning reproduction outside these terms should be sent to the publishers at the undermentioned address:

ISTE Ltd
6 Fitzroy Square
London W1T 5DX
UK

ISTE USA
4308 Patrice Road
Newport Beach, CA 92663
USA

www.iste.co.uk

© ISTE Ltd, 2007

© LAVOISIER, 2001, 2002

The rights of Etienne Dombre and Wisama Khalil to be identified as the authors of this work have been asserted by them in accordance with the Copyright, Designs and Patents Act 1988.

Library of Congress Cataloging-in-Publication Data

Modeling, performance analysis and control of robot manipulators/edited by Etienne Dombre, Wisama Khalil.

p. cm.

Includes index.

ISBN-13: 978-1-905209-10-1

ISBN-10: 1-905209-10-X

1. Robotics. 2. Manipulators (Mechanism) I. Dombre, E. (Etienne) II. Khalil, W. (Wisama)
TJ211.M626 2006

629.8'933--dc22

2006032328

British Library Cataloguing-in-Publication Data

A CIP record for this book is available from the British Library

ISBN 10: 1-905209-10-X

ISBN 13: 978-1-905209-10-1

Printed and bound in Great Britain by Antony Rowe Ltd, Chippenham, Wiltshire.

Table of Contents

Chapter 1. Modeling and Identification of Serial Robots.	1
Wisama KHALIL and Etienne DOMBRE	
1.1. Introduction.	1
1.2. Geometric modeling.	2
1.2.1. Geometric description	2
1.2.2. Direct geometric model	6
1.2.3. Inverse geometric model.	7
1.2.3.1. Stating the problem.	8
1.2.3.2. Principle of Paul's method	10
1.3. Kinematic modeling.	14
1.3.1. Direct kinematic model	14
1.3.1.1 Calculation of the Jacobian matrix by derivation of the DGM	15
1.3.1.2. Kinematic Jacobian matrix	17
1.3.1.3. Decomposition of the kinematic Jacobian matrix into three matrices	19
1.3.1.4. Dimension of the operational space of a robot.	20
1.3.2. Inverse kinematic model.	21
1.3.2.1. General form of the kinematic model	21
1.3.2.2. Inverse kinematic model for the regular case	22
1.3.2.3. Solution at the proximity of singular positions	23
1.3.2.4. Inverse kinematic model of redundant robots	24

1.4. Calibration of geometric parameters	26
1.4.1. Introduction	26
1.4.2. Geometric parameters	26
1.4.2.1. Geometric parameters of the robot.	26
1.4.2.2. Parameters of the robot's location	27
1.4.2.3. Geometric parameters of the end-effector	28
1.4.3. Generalized differential model of a robot.	29
1.4.4. Principle of geometric calibration	30
1.4.4.1. General form of the calibration model	30
1.4.4.2. Identifying the geometric parameters	31
1.4.4.3. Solving the identification equations	34
1.4.5. Calibration methods of geometric parameters	35
1.4.5.1. Calibration model by measuring the end-effector location.	35
1.4.5.2. Autonomous calibration models	36
1.4.6. Correction of geometric parameters	39
1.5. Dynamic modeling	40
1.5.1. Lagrange formalism	42
1.5.1.1. General form of dynamic equations	43
1.5.1.2. Calculation of energy	44
1.5.1.3. Properties of the dynamic model.	46
1.5.1.4. Taking into consideration the friction	47
1.5.1.5. Taking into account the inertia of the actuator's rotor	48
1.5.1.6. Taking into consideration the forces and moments exerted by the end-effector on its environment	48
1.5.2. Newton-Euler formalism	50
1.5.2.1. Newton-Euler equations linear in the inertial parameters	50
1.5.2.2. Practical form of Newton-Euler equations	52
1.5.3. Determining the base inertial parameters	53
1.6. Identification of dynamic parameters.	59
1.6.1. Introduction	59
1.6.2. Identification principle of dynamic parameters	60
1.6.2.1. Solving method	60
1.6.2.2. Identifiable parameters.	62
1.6.2.3. Choice of identification trajectories	63
1.6.2.4. Evaluation of joint coordinates	65
1.6.2.5. Evaluation of joint torques	65

1.6.3. Identification model using the dynamic model	66
1.6.4. Sequential formulation of the dynamic model	68
1.6.5. Practical considerations	69
1.7. Conclusion	70
1.8. Bibliography	71
Chapter 2. Modeling of Parallel Robots	81
Jean-Pierre MERLET and François PIERROT	
2.1. Introduction.	81
2.1.1. Characteristics of classic robots	81
2.1.2. Other types of robot structure.	82
2.1.3. General advantages and disadvantages	86
2.1.4. Present day uses.	88
2.1.4.1. Simulators and space applications	88
2.1.4.2. Industrial applications	91
2.1.4.3. Medical applications	93
2.1.4.4. Precise positioning	94
2.2. Machine types	95
2.2.1. Introduction	95
2.2.2. Plane robots with three degrees of freedom	100
2.2.3. Robots moving in space	101
2.2.3.1. Manipulators with three degrees of freedom	101
2.2.3.2. Manipulators with four or five degrees of freedom	107
2.2.3.3. Manipulators with six degrees of freedom	109
2.3. Inverse geometric and kinematic models	113
2.3.1. Inverse geometric model.	113
2.3.2. Inverse kinematics	115
2.3.3. Singular configurations	117
2.3.3.1. Singularities and statics	121
2.3.3.2. State of the art.	121
2.3.3.3. The geometric method.	122
2.3.3.4. Maneuverability and condition number.	125
2.3.3.5. Singularities in practice	126

2.4. Direct geometric model.	126
2.4.1. Iterative method.	127
2.4.2. Algebraic method.	128
2.4.2.1. Reminder concerning algebraic geometry	128
2.4.2.2. Planar robots.	130
2.4.2.3. Manipulators with six degrees of freedom	133
2.5. Bibliography	134
Chapter 3. Performance Analysis of Robots.	141
Philippe WENGER	
3.1. Introduction.	141
3.2. Accessibility	143
3.2.1. Various levels of accessibility	143
3.2.2. Condition of accessibility	144
3.3. Workspace of a robot manipulator	146
3.3.1. General definition.	146
3.3.2. Space of accessible positions	148
3.3.3. Primary space and secondary space	149
3.3.4. Defined orientation workspace	151
3.3.5. Free workspace	152
3.3.6. Calculation of the workspace	155
3.4. Concept of aspect	157
3.4.1. Definition	157
3.4.2. Mode of aspects calculation.	158
3.4.3. Free aspects	160
3.4.4. Application of the aspects	161
3.5. Concept of connectivity	163
3.5.1. Introduction	163
3.5.2. Characterization of n-connectivity	165
3.5.3. Characterization of t-connectivity	168
3.6. Local performances	174
3.6.1. Definition of dexterity	174
3.6.2. Manipulability.	174
3.6.3. Isotropy index	180

3.6.4. Lowest singular value	181
3.6.5. Approach lengths and angles	181
3.7. Conclusion	183
3.8. Bibliography	183
Chapter 4. Trajectory Generation	189
Moussa HADDAD, Taha CHETTIBI, Wisama KHALIL and Halim LEHTIHET	
4.1. Introduction.	189
4.2. Point-to-point trajectory in the joint space under kinematic constraints	190
4.2.1. Fifth-order polynomial model	191
4.2.2. Trapezoidal velocity model	193
4.2.3. Smoothed trapezoidal velocity model	198
4.3. Point-to-point trajectory in the task-space under kinematic constraints	201
4.4. Trajectory generation under kinodynamic constraints.	204
4.4.1. Problem statement	205
4.4.1.1. Constraints.	206
4.4.1.2. Objective function	207
4.4.2. Description of the method.	208
4.4.2.1. Outline	208
4.4.2.2. Construction of a random trajectory profile	209
4.4.2.3. Handling kinodynamic constraints.	212
4.4.2.4. Summary	216
4.4.3. Trapezoidal profiles	218
4.5. Examples	221
4.5.1. Case of a two dof robot	221
4.5.1.1. Optimal free motion planning problem	221
4.5.1.2. Optimal motion problem with geometric path constraint.	223
4.5.2. Case of a six dof robot.	224
4.5.2.1. Optimal free motion planning problem	225
4.5.2.2. Optimal motion problem with geometric path constraints	226

4.5.2.3. Optimal free motion planning problem with intermediate points	227
4.6. Conclusion	229
4.7. Bibliography	230
Appendix: Stochastic Optimization Techniques	234

Chapter 5. Position and Force Control of a Robot in a Free or Constrained Space. 241

Pierre DAUCHEZ and Philippe FRAISSE

5.1. Introduction.	241
5.2. Free space control	242
5.2.1. Hypotheses applying to the whole chapter	242
5.2.2. Complete dynamic modeling of a robot manipulator	243
5.2.3. Ideal dynamic control in the joint space	246
5.2.4. Ideal dynamic control in the operational working space	248
5.2.5. Decentralized control	250
5.2.6. Sliding mode control	251
5.2.7. Robust control based on high order sliding mode	254
5.2.8. Adaptive control	255
5.3. Control in a constrained space.	257
5.3.1. Interaction of the manipulator with the environment	257
5.3.2. Impedance control	257
5.3.3. Force control of a mass attached to a spring	258
5.3.4. Non-linear decoupling in a constrained space	262
5.3.5. Position/force hybrid control	263
5.3.5.1. Parallel structure	263
5.3.5.2. External structure	269
5.3.6. Specificity of the force/torque control.	271
5.4. Conclusion	275
5.5. Bibliography	275

Chapter 6. Visual Servoing	279
François CHAUMETTE	
6.1. Introduction.	279
6.2. Modeling visual features	281
6.2.1. The interaction matrix	281
6.2.2. Eye-in-hand configuration.	282
6.2.3. Eye-to-hand configuration.	283
6.2.4. Interaction matrix	284
6.2.4.1. Interaction matrix of a 2-D point	284
6.2.4.2. Interaction matrix of a 2-D geometric primitive	287
6.2.4.3. Interaction matrix for complex 2-D shapes	290
6.2.4.4. Interaction matrix by learning or estimation	293
6.2.5. Interaction matrix related to 3-D visual features.	294
6.2.5.1. Pose estimation	294
6.2.5.2. Interaction matrix related to θ_u	297
6.2.5.3. Interaction matrix related to a 3-D point	298
6.2.5.4. Interaction matrix related to a 3-D plane	300
6.3. Task function and control scheme	301
6.3.1. Obtaining the desired value s^*	301
6.3.2. Regulating the task function	302
6.3.2.1. Case where the dimension of s is 6 ($k = 6$)	304
6.3.2.2. Case where the dimension of s is greater than 6 ($k > 6$)	312
6.3.3. Hybrid tasks	317
6.3.3.1. Virtual links	317
6.3.3.2. Hybrid task function	319
6.3.4. Target tracking	323
6.4. Other exteroceptive sensors	325
6.5. Conclusion	326
6.6. Bibliography	328

Chapter 7. Modeling and Control of Flexible Robots 337

Frédéric BOYER, Wisama KHALIL, Mouhacine BENOSMAN and
George LE VEY

7.1. Introduction.	337
7.2. Modeling of flexible robots	337
7.2.1. Introduction	337
7.2.2. Generalized Newton-Euler model for a kinematically free elastic body	339
7.2.2.1. Definition: formalism of a dynamic model.	339
7.2.2.2. Choice of formalism	340
7.2.2.3. Kinematic model of a free elastic body	341
7.2.2.4. Balance principle compatible with the mixed formalism.	343
7.2.2.5. Virtual power of the field of acceleration quantities	344
7.2.2.6. Virtual power of external forces	346
7.2.2.7. Virtual power of elastic cohesion forces	347
7.2.2.8. Balance of virtual powers	348
7.2.2.9. Linear rigid balance in integral form	349
7.2.2.10. Angular rigid balance in integral form.	349
7.2.2.11. Elastic balances in integral form	350
7.2.2.12. Linear rigid balance in parametric form.	351
7.2.2.13. Intrinsic matrix form of the generalized Newton-Euler model.	353
7.2.3. Velocity model of a simple open robotic chain	356
7.2.4. Acceleration model of a simple open robotic chain.	357
7.2.5. Generalized Newton-Euler model for a flexible manipulator	358
7.2.6. Extrinsic Newton-Euler model for numerical calculus	359
7.2.7. Geometric model of an open chain.	362
7.2.8. Recursive calculation of the inverse and direct dynamic models for a flexible robot	363
7.2.8.1. Introduction	363
7.2.8.2. Recursive algorithm of the inverse dynamic model.	364
7.2.8.3. Recursive algorithm of the direct dynamic model.	368
7.2.8.4. Iterative symbolic calculation.	373
7.3. Control of flexible robot manipulators	373
7.3.1. Introduction	373
7.3.2. Reminder of notations	374

7.3.3. Control methods	375
7.3.3.1. Regulation	375
7.3.3.2. Point-to-point movement in fixed time	375
7.3.3.3. Trajectory tracking in the joint space	380
7.3.3.4. Trajectory tracking in the operational space	383
7.4. Conclusion	388
7.5. Bibliography	389
List of Authors	395
Index	397

This page intentionally left blank

Chapter 1

Modeling and Identification of Serial Robots

1.1. Introduction

The design and control of robots require certain mathematical models, such as:

– transformation models between the operational space (in which the position of the end-effector is defined) and the joint space (in which the configuration of the robot is defined). The following is distinguished:

- direct and inverse geometric models giving the location of the end-effector (or the tool) in terms of the joint coordinates of the mechanism and vice versa,

- direct and inverse kinematic models giving the velocity of the end-effector in terms of the joint velocities and vice versa,

– dynamic models giving the relations between the torques or forces of the actuators, and the positions, velocities and accelerations of the joints.

This chapter presents some methods to establish these models. It will also deal with identifying the parameters appearing in these models. We will limit the discussion to simple open structures. For complex structure robots, i.e. tree or closed structures, we refer the reader to [KHA 02].

Mathematical development is based on (4×4) homogenous transformation matrices. The homogenous matrix ${}^i\mathbf{T}_j$ representing the transformation from frame R_i to frame R_j is defined as:

$${}^i\mathbf{T}_j = \begin{bmatrix} {}^i\mathbf{R}_j & {}^i\mathbf{P}_j \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^i\mathbf{s}_j & {}^i\mathbf{n}_j & {}^i\mathbf{a}_j & {}^i\mathbf{P}_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [1.1]$$

where ${}^i\mathbf{s}_j$, ${}^i\mathbf{n}_j$ and ${}^i\mathbf{a}_j$ of the orientation matrix ${}^i\mathbf{R}_j$ indicate the unit vectors along the axes \mathbf{x}_j , \mathbf{y}_j and \mathbf{z}_j of the frame R_j expressed in the frame R_i ; and where ${}^i\mathbf{P}_j$ is the vector expressing the origin of the frame R_j in the frame R_i .

1.2. Geometric modeling

1.2.1. Geometric description

A systematic and automatic modeling of robots requires an appropriate method for the description of their morphology. Several methods and notations have been proposed [DEN 55], [SHE 71], [REN 75], [KHA 76], [BOR 79], [CRA 86]. The most widely used one is that of Denavit-Hartenberg [DEN 55]. However, this method, developed for simple open structures, presents ambiguities when it is applied to closed or tree-structured robots. Hence, we recommend the notation of Khalil and Kleinfinger which enables the unified description of complex and serial structures of articulated mechanical systems [KHA 86].

A simple open structure consists of $n+1$ links noted C_0, \dots, C_n and of n joints. Link C_0 indicates the robot base and link C_n , the link carrying the end-effector. Joint j connects link C_j to link C_{j-1} (Figure 1.1). The method of description is based on the following rules and conventions:

- the links are assumed to be perfectly rigid. They are connected by revolute or prismatic joints considered as being ideal (no mechanical clearance, no elasticity);
- the frame R_j is fixed to link C_j ;
- axis \mathbf{z}_j is along the axis of joint j ;
- axis \mathbf{x}_j is along the common perpendicular with axes \mathbf{z}_j and \mathbf{z}_{j+1} . If axes \mathbf{z}_j and \mathbf{z}_{j+1} are parallel or collinear, the choice of \mathbf{x}_j is not unique: considerations of symmetry or simplicity lead to a reasonable choice.

The transformation matrix from the frame R_{j-1} to the frame R_j is expressed in terms of the following four geometric parameters:

- α_j : angle between axes \mathbf{z}_{j-1} and \mathbf{z}_j corresponding to a rotation about \mathbf{x}_{j-1} ;

- d_j : distance between z_{j-1} and z_j along x_{j-1} ;
- θ_j : angle between axes x_{j-1} and x_j corresponding to a rotation about z_j ;
- r_j : distance between x_{j-1} and x_j along z_j .

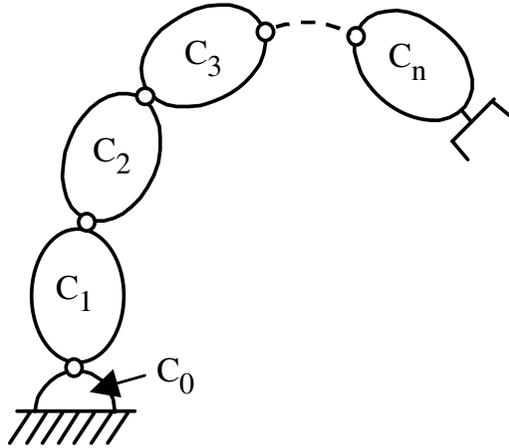


Figure 1.1. A simple open structure robot

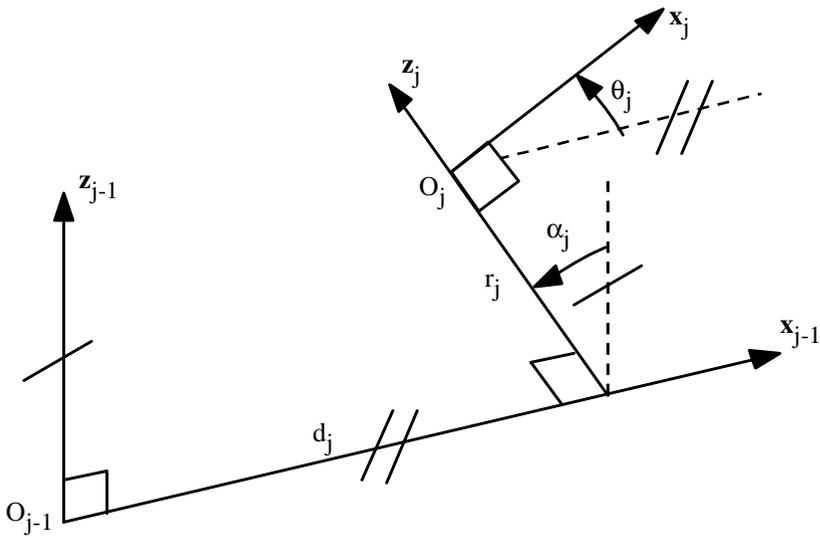


Figure 1.2. Geometric parameters in the case of a simple open structure

The joint coordinate q_j associated to the j^{th} joint is either θ_j or r_j , depending on whether this joint is revolute or prismatic. It can be expressed by the relation:

$$q_j = \bar{\sigma}_j \theta_j + \sigma_j r_j \quad [1.2]$$

with:

- $\sigma_j = 0$ if the joint is revolute;
- $\sigma_j = 1$ if the joint is prismatic;
- $\bar{\sigma}_j = 1 - \sigma_j$.

The transformation matrix defining the frame R_j in the frame R_{j-1} is obtained from Figure 1.2 by:

$${}^{j-1}T_j = \mathbf{Rot}(\mathbf{x}, \alpha_j) \mathbf{Trans}(\mathbf{x}, d_j) \mathbf{Rot}(\mathbf{z}, \theta_j) \mathbf{Trans}(\mathbf{z}, r_j)$$

$$= \begin{bmatrix} C\theta_j & -S\theta_j & 0 & d_j \\ C\alpha_j S\theta_j & C\alpha_j C\theta_j & -S\alpha_j & -r_j S\alpha_j \\ S\alpha_j S\theta_j & S\alpha_j C\theta_j & C\alpha_j & r_j C\alpha_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [1.3]$$

where $\mathbf{Rot}(\mathbf{u}, \alpha)$ and $\mathbf{Trans}(\mathbf{u}, d)$ are (4×4) homogenous matrices representing, respectively, a rotation α about the axis \mathbf{u} and a translation d along \mathbf{u} .

NOTES.

– for the definition of the reference frame R_0 , the simplest choice consists of taking R_0 aligned with the frame R_1 when $q_1 = 0$, which indicates that \mathbf{z}_0 is along \mathbf{z}_1 and $O_0 \equiv O_1$ when joint 1 is revolute, and \mathbf{z}_0 is along \mathbf{z}_1 and \mathbf{x}_0 is parallel to \mathbf{x}_1 when joint 1 is prismatic. This choice renders the parameters α_1 and d_1 zero;

– likewise, the axis \mathbf{x}_n of the frame R_n is taken collinear to \mathbf{x}_{n-1} when $q_n = 0$. This choice makes r_n (or θ_n) zero when $\sigma_n = 1$ (or = 0 respectively);

– for a prismatic joint, the axis \mathbf{z}_j is parallel to the axis of the joint; it can be placed in such a way that d_j or d_{j+1} is zero;

– when \mathbf{z}_j is parallel to \mathbf{z}_{j+1} , the axis \mathbf{x}_j is placed in such a way that r_j or r_{j+1} is zero;

– in practice, the vector of joint variables \mathbf{q} is given by:

$$\mathbf{q} = \mathbf{K}_c \mathbf{q}_c + \mathbf{q}_0$$

where \mathbf{q}_0 represents an offset, \mathbf{q}_c are encoder variables and \mathbf{K}_c is a constant matrix.

EXAMPLE 1.1.– description of the structure of the Stäubli RX-90 robot (Figure 1.3). The robot shoulder is of an RRR anthropomorphic type and the wrist consists of three intersecting revolute axes, equivalent to a spherical joint. From a methodological point of view, firstly the axes \mathbf{z}_j are placed on the joint axes and the axes \mathbf{x}_j are placed according to the rules previously set. Next, the geometric parameters of the robot are determined. The link frames are shown in Figure 1.3 and the geometric parameters are given in Table 1.1.

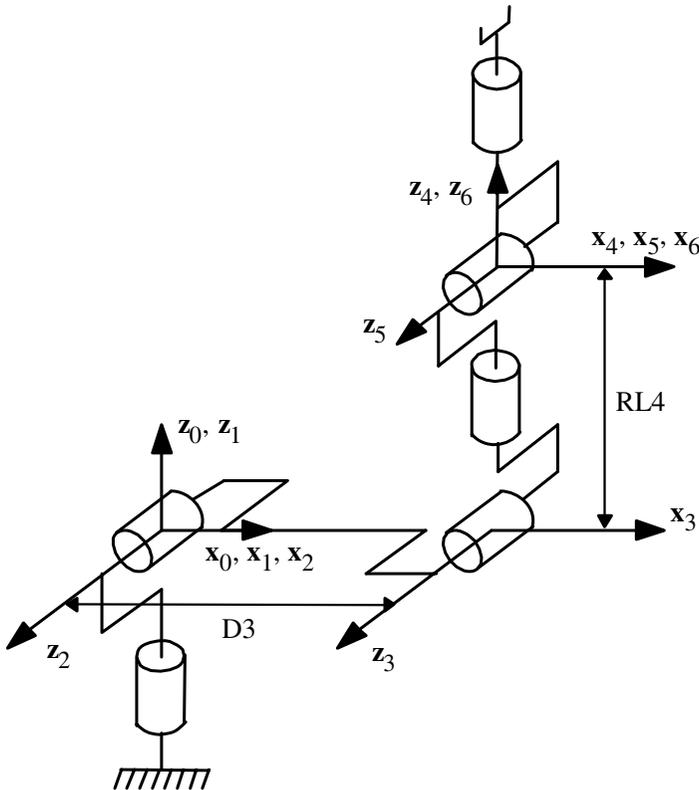


Figure 1.3. Link frames for the Stäubli RX-90 robot

j	σ_j	α_j	d_j	θ_j	r_j
1	0	0	0	θ_1	0
2	0	$\pi/2$	0	θ_2	0
3	0	0	D3	θ_3	0
4	0	$-\pi/2$	0	θ_4	RL4
5	0	$\pi/2$	0	θ_5	0
6	0	$-\pi/2$	0	θ_6	0

Table 1.1. Geometric parameters for the Stäubli RX-90 robot

1.2.2. Direct geometric model

The direct geometric model (DGM) represents the relations calculating the operational coordinates, giving the location of the end-effector, in terms of the joint coordinates. In the case of a simple open chain, it can be represented by the transformation matrix ${}^0\mathbf{T}_n$:

$${}^0\mathbf{T}_n = {}^0\mathbf{T}_1(q_1) {}^1\mathbf{T}_2(q_2) \dots {}^{n-1}\mathbf{T}_n(q_n) \quad [1.4]$$

The direct geometric model of the robot may also be represented by the relation:

$$\mathbf{X} = \mathbf{f}(\mathbf{q}) \quad [1.5]$$

\mathbf{q} being the vector of joint coordinates such that:

$$\mathbf{q} = [q_1 \ q_2 \ \dots \ q_n]^T \quad [1.6]$$

The operational coordinates are defined by:

$$\mathbf{X} = [x_1 \ x_2 \ \dots \ x_m]^T \quad [1.7]$$

There are several possibilities to define the vector \mathbf{X} . For example, with the help of the elements of matrix ${}^0\mathbf{T}_n$:

$$\mathbf{X} = [P_x \ P_y \ P_z \ s_x \ s_y \ s_z \ n_x \ n_y \ n_z \ a_x \ a_y \ a_z]^T \quad [1.8]$$

or otherwise, knowing that $\mathbf{s} = \mathbf{n}\mathbf{x}\mathbf{a}$

$$\mathbf{X} = [P_x \ P_y \ P_z \ n_x \ n_y \ n_z \ a_x \ a_y \ a_z]^T \quad [1.9]$$

For the orientation, other representations are currently used such as Euler angles, Roll-Pitch-Yaw angles or Quaternions. We can easily derive direction cosines \mathbf{s} , \mathbf{n} , \mathbf{a} from any one of these representations and vice versa [KHA 02].

EXAMPLE 1.2. – direct geometric model for the Stäubli RX-90 robot (Figure 1.3). According to Table 1.1, the relation [1.3] can be used to write the basic transformation matrices ${}^j-1\mathbf{T}_j$. The product of these matrices gives ${}^0\mathbf{T}_6$ that has as components:

$$\begin{aligned} s_x &= C1(C23(C4C5C6 - S4S6) - S23S5C6) - S1(S4C5C6 + C4S6) \\ s_y &= S1(C23(C4C5C6 - S4S6) - S23S5C6) + C1(S4C5C6 + C4S6) \\ s_z &= S23(C4C5C6 - S4S6) + C23S5C6 \\ n_x &= C1(-C23(C4C5S6 + S4C6) + S23S5S6) + S1(S4C5S6 - C4C6) \\ n_y &= S1(-C23(C4C5S6 + S4C6) + S23S5S6) - C1(S4C5S6 - C4C6) \\ n_z &= -S23(C4C5S6 + S4C6) - C23S5S6 \\ a_x &= -C1(C23C4S5 + S23C5) + S1S4S5 \\ a_y &= -S1(C23C4S5 + S23C5) - C1S4S5 \\ a_z &= -S23C4S5 + C23C5 \\ P_x &= -C1(S23 RL4 - C2D3) \\ P_y &= -S1(S23 RL4 - C2D3) \\ P_z &= C23 RL4 + S2D3 \end{aligned}$$

with $C23 = \cos(\theta_2 + \theta_3)$ and $S23 = \sin(\theta_2 + \theta_3)$.

1.2.3. Inverse geometric model

We saw that the direct geometric model of a robot calculates the operational coordinates giving the location of the end-effector in terms of joint coordinates. The

inverse problem consists of calculating the joint coordinates corresponding to a given location of the end-effector. When it exists, the explicit form which gives all possible solutions (there is rarely uniqueness of solution) constitutes what we call the inverse geometric model (IGM). We can distinguish three methods for the calculation of IGM:

- Paul’s method [PAU 81], which deals with each robot separately and is suitable for most of the industrial robots;
- Pieper’s method [PIE 68], which makes it possible to solve the problem for the robots with six degrees of freedom having three revolute joints with intersecting axes or three prismatic joints;
- the general Raghavan and Roth’s method [RAG 90] giving the general solution for robots with six joints using at most a 16-degree polynomial.

Whenever calculating an explicit form of the inverse geometric model is not possible, we can calculate a particular solution through numeric procedures [PIE 68], [WHI 69], [FOU 80], [FEA 83], [WOL 84], [GOL 85] [SCI 86].

In this chapter, we present Paul’s method; Pieper’s method, and Raghavan and Roth’s method are detailed in [KHA 02].

1.2.3.1. *Stating the problem*

Let ${}^f\mathbf{T}_E^d$ be the homogenous transformation matrix representing the desired location of the end-effector frame R_E with respect to the world frame R_f . In general cases, ${}^f\mathbf{T}_E^d$ can be expressed in the following form:

$${}^f\mathbf{T}_E^d = \mathbf{Z} {}^0\mathbf{T}_n(\mathbf{q}) \mathbf{E} \quad [1.10]$$

where (see Figure 1.4):

- \mathbf{Z} is the transformation matrix defining the location of the robot frame R_0 in the world reference frame R_f ;
- ${}^0\mathbf{T}_n$ is the transformation matrix of the terminal link frame R_n with respect to frame R_0 in terms of the joint coordinates \mathbf{q} ;
- \mathbf{E} is the transformation matrix defining the end-effector frame R_E in the terminal frame R_n .

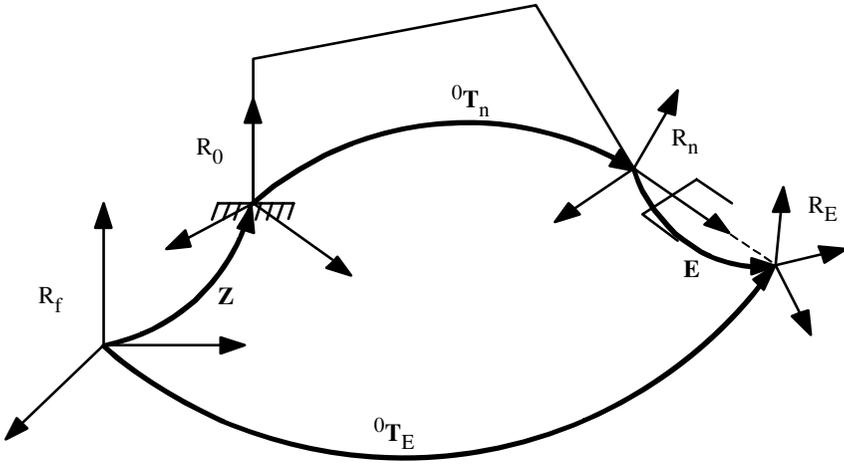


Figure 1.4. Transformations between the end-effector frame and the world reference frame

When $n \geq 6$, we can write the following relation by grouping on the right hand side all known terms:

$${}^0T_n(\mathbf{q}) = \mathbf{Z}^{-1} {}^fT_E^d \mathbf{E}^{-1} \quad [1.11]$$

When $n < 6$, the robot's operational space is less than six. It is not possible to place the end-effector frame R_E in an arbitrary location R_E^d describing the task, except when the frames R_E and R_E^d are conditioned in a particular way in order to compensate for the insufficient number of degrees of freedom. Practically, instead of bringing frame R_E onto frame R_E^d , we will seek to only place some elements of the end-effector (points, straight lines).

In the calculation of IGM, three cases can be distinguished:

a) no solution when the desired location is outside of the accessible zone of the robot. It is limited by the number of degrees of freedom of the robot, the joint limits and the dimension of the links;

b) infinite number of solutions when:

- the robot is redundant with respect to the task,
- the robot is in some singular configuration;

c) a finite number of solutions expressed by a set of vectors $\{\mathbf{q}^1, \dots, \mathbf{q}^r\}$. A robot is said to be solvable [PIE 68], [ROT 76] when it is possible to calculate all the

configurations making it possible to reach a given location. Nowadays, all serial manipulators having up to six degrees of freedom and which are not redundant may be considered as solvable. The number of solutions depends on the structure of the robot and is at most equal to 16.

1.2.3.2. Principle of Paul's method

Let us consider a robot whose homogenous transformation matrix has the following form:

$${}^0\mathbf{T}_n = {}^0\mathbf{T}_1(q_1) {}^1\mathbf{T}_2(q_2) \dots {}^{n-1}\mathbf{T}_n(q_n) \quad [1.12]$$

Let \mathbf{U}_0 be the desired location, such that:

$$\mathbf{U}_0 = \begin{bmatrix} s_x & n_x & a_x & P_x \\ s_y & n_y & a_y & P_y \\ s_z & n_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [1.13]$$

We seek to solve the following system of equations:

$$\mathbf{U}_0 = {}^0\mathbf{T}_1(q_1) {}^1\mathbf{T}_2(q_2) \dots {}^{n-1}\mathbf{T}_n(q_n) \quad [1.14]$$

Paul's method consists of successively pre-multiplying the two sides of equation [1.14] by the matrices ${}^j\mathbf{T}_{j-1}$ for $j = 1, \dots, n-1$, operations which make it possible to isolate and identify one after another of the joint coordinates.

For example, in the case of a six degrees of freedom robot, the procedure is as follows:

– left multiplication of both sides of equation [1.14] by ${}^1\mathbf{T}_0$:

$${}^1\mathbf{T}_0 \mathbf{U}_0 = {}^1\mathbf{T}_2 {}^2\mathbf{T}_3 {}^3\mathbf{T}_4 {}^4\mathbf{T}_5 {}^5\mathbf{T}_6 \quad [1.15]$$

The right hand side is a function of the variables q_2, \dots, q_6 . The left hand side is only a function of the variable q_1 ;

– term-to-term identification of the two sides of equation [1.15]. We obtain a system of one or two equations function of q_1 only, whose structure belongs to a particular type amongst a dozen of possible types;

– left multiplication of both sides of equation [1.15] by ${}^2\mathbf{T}_1$ and calculation of q_2 .

The succession of equations enabling the calculation of all q_j is the following:

$$\begin{aligned}
 \mathbf{U}_0 &= {}^0\mathbf{T}_1 {}^1\mathbf{T}_2 {}^2\mathbf{T}_3 {}^3\mathbf{T}_4 {}^4\mathbf{T}_5 {}^5\mathbf{T}_6 \\
 {}^1\mathbf{T}_0 \mathbf{U}_0 &= {}^1\mathbf{T}_2 {}^2\mathbf{T}_3 {}^3\mathbf{T}_4 {}^4\mathbf{T}_5 {}^5\mathbf{T}_6 \\
 {}^2\mathbf{T}_1 \mathbf{U}_1 &= {}^2\mathbf{T}_3 {}^3\mathbf{T}_4 {}^4\mathbf{T}_5 {}^5\mathbf{T}_6 & [1.16] \\
 {}^3\mathbf{T}_2 \mathbf{U}_2 &= {}^3\mathbf{T}_4 {}^4\mathbf{T}_5 {}^5\mathbf{T}_6 \\
 {}^4\mathbf{T}_3 \mathbf{U}_3 &= {}^4\mathbf{T}_5 {}^5\mathbf{T}_6 \\
 {}^5\mathbf{T}_4 \mathbf{U}_4 &= {}^5\mathbf{T}_6
 \end{aligned}$$

with:

$$\mathbf{U}_{j+1} = {}^{j+1}\mathbf{T}_6 = {}^{j+1}\mathbf{T}_j \mathbf{U}_j \text{ for } j = 0, \dots, 4 \quad [1.17]$$

The use of this method for a large number of industrial robots has shown that only a few types of equations are encountered, and that their solutions are relatively simple.

NOTES.

1) When a robot has more than six degrees of freedom, the system to be solved contains more unknowns than parameters describing the task: it lacks $(n-6)$ relations. Two strategies are possible:

– the first strategy consists of setting arbitrarily $(n-6)$ joint variables. In this case we deal with a problem with six degrees of freedom. The choice of these joints results from the task's specifications and from the structure of the robot;

– the second strategy consists of introducing $(n-6)$ supplementary relations describing the redundancy, like for example in [HOL 84] for robots with seven degrees of freedom.

2) A robot with less than six degrees of freedom cannot place its end-effector at arbitrary positions and orientations. Thus, it is not possible to bring the end-effector frame \mathbf{R}_E onto another desired frame \mathbf{R}_E^d except if certain elements of ${}^0\mathbf{T}_E^d$ are imposed in a way that compensates for the insufficient number of degrees of freedom. Otherwise, we have to reduce the number of equations by considering only certain elements (points or axes) of the frames \mathbf{R}_E and \mathbf{R}_E^d .

EXAMPLE 1.3.– inverse geometric model of the Stäubli RX-90 robot. After performing all the calculations, we obtain the following solutions:

$$\begin{cases} \theta_1 = \text{atan2}(P_y, P_x) \\ \theta'_1 = \theta_1 + \pi \end{cases}$$

$$\theta_2 = \text{atan2}(S2, C2)$$

with:

$$\begin{cases} C2 = \frac{YZ - \varepsilon X \sqrt{X^2 + Y^2 - Z^2}}{X^2 + Y^2} \\ S2 = \frac{XZ - \varepsilon Y \sqrt{X^2 + Y^2 - Z^2}}{X^2 + Y^2} \end{cases} \quad \text{with } \varepsilon = \pm 1$$

$$X = -2P_z D3$$

$$Y = -2 B1 D3$$

$$Z = (RL4)^2 - (D3)^2 - (P_z)^2 - (B1)^2$$

$$B1 = P_x C1 + P_y S1$$

$$\theta_3 = \text{atan2}\left(\frac{-P_z S2 - B1 C2 + D3}{RL4}, \frac{-B1 S2 + P_z C2}{RL4}\right)$$

$$\begin{cases} \theta_4 = \text{atan2}[S1 a_x - C1 a_y, -C23(C1 a_x + S1 a_y) - S23 a_z] \\ \theta'_4 = \theta_4 + \pi \end{cases}$$

$$\theta_5 = \text{atan2}(S5, C5)$$

with:

$$S5 = -C4 [C23 (C1 a_x + S1 a_y) + S23 a_z] + S4 (S1 a_x - C1 a_y)$$

$$C5 = -S23 (C1 a_x + S1 a_y) + C23 a_z$$

$$\theta_6 = \text{atan2}(S6, C6)$$

with:

$$S6 = -C4 (S1 s_x - C1 s_y) - S4 [C23 (C1 s_x + S1 s_y) + S23 s_z]$$

$$C6 = -C4 (S1 n_x - C1 n_y) - S4 [C23 (C1 n_x + S1 n_y) + S23 n_z]$$

NOTES.

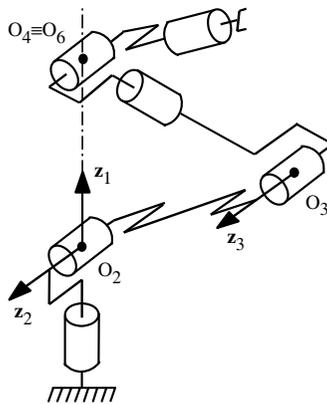
1) Singular positions:

i) when $P_x = P_y = 0$, which corresponds to $S23RL4 - C2D3 = 0$, the point O_4 is on the axis z_0 (Figure 1.5a). The two arguments used for calculating θ_1 are zero and consequently θ_1 is not determined. We can give any value to θ_1 , generally the value of the current position, or, according to optimization criteria, such as maximizing the distance from the mechanical limits of the joints. This means that we can always find a solution, but a small change of the desired position might call for a significant variation of θ_1 , which may be impossible to carry out considering the velocity and acceleration limits of the actuators,

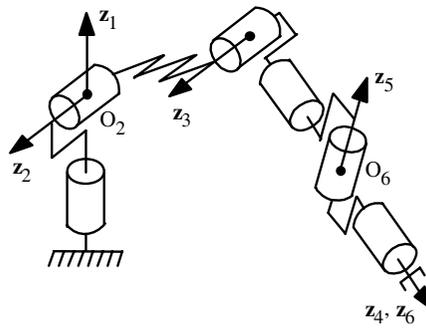
ii) when $C23(C1a_x + S1a_y) + S23a_z = 0$ and $S1a_x - C1a_y = 0$, the two arguments of the atan2 function used for the calculation of θ_4 are zero and hence the function is not determined. This configuration happens when axes 4 and 6 are aligned ($C\theta_5 = \pm 1$) and it is the sum ($\theta_4 \pm \theta_6$) which can be obtained (see Figure 1.5b). We can give to θ_4 its current value, then we calculate θ_6 according to this value. We can also calculate the values of θ_4 and θ_6 , which move joints 4 and 6 away from their limits,

iii) a third singular position occurring when $C3 = 0$ will be highlighted along with the kinematic model. This singularity does not pose any problem for the inverse geometric model (see Figure 1.5c).

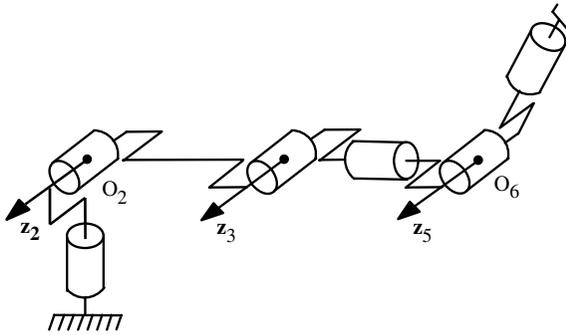
2) Number of solutions: apart from singularities, the Stäubli RX-90 robot has eight theoretical configurations for the IGM (product of the number of possible solutions on each axis). Some of these configurations may not be accessible due to their joint limits.



a) Singularity of the shoulder ($P_x = P_y = 0$ and $S23RL4 - C2D3 = 0$)



b) Singularity of the wrist ($S5 = 0$)



c) Singularity of the elbow ($C3 = 0$)

Figure 1.5. Singular positions of the Stäubli RX-90 robot

1.3. Kinematic modeling

1.3.1. Direct kinematic model

The direct kinematic model of a robot gives the velocities of the operational coordinates $\dot{\mathbf{X}}$ in terms of the joint velocities $\dot{\mathbf{q}}$. We write:

$$\dot{\mathbf{X}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \tag{1.18}$$