Mobile Robotics

Luc Jaulin

ISTE PRESS

ELSEVIER

Series Editor
Hisham Abou Kandil

Mobile Robotics

Luc Jaulin

iSTE PRESS

ELSEVIER

First published 2015 in Great Britain and the United States by ISTE Press Ltd and Elsevier Ltd

ISTE Press Ltd
27-37 St George's Road
London SW19 4EU
UK

www.iste.co.uk

Elsevier Ltd
The Boulevard, Langford Lane
Kidlington, Oxford, OX5 1GB
UK

www.elsevier.com

British Library Cataloguing-in-Publication Data
A CIP record for this book is available from the British Library
Library of Congress Cataloging in Publication Data
A catalog record for this book is available from the Library of Congress
ISBN 978-1-78548-048-5

Printed and bound in the UK and US

Mobile Robotics

Introduction

A *mobile robot* can be defined as a mechanical system capable of moving in its environment in an autonomous manner. For that purpose, it must be equipped with:

– *sensors* that will help in gaining knowledge of its surroundings (which it is more or less aware of) and determine its location;

– *actuators* which will allow it to move;

– an *intelligence* (or algorithm, regulator), which will allow it to compute, based on the data gathered by the sensors, the commands to send to the actuators in order to perform a given task.

Finally, to this we must add the *surroundings* of the robot which correspond to the world in which it evolves and its *mission* which is the task it has to accomplish. Mobile robots have been constantly evolving, mainly from the beginning of the 2000s, in military domains (airborne drones [BEA 12], underwater robots [CRE 14], etc.), and even in medical and agricultural fields. They are in particularly high demand for performing tasks considered to be painful or dangerous to humans. This is the case, for instance, in mine-clearing operations, the search for black boxes of damaged aircraft on the ocean bed and planetary exploration. Artificial satellites, launchers (such as Ariane V), driverless subways and elevators are examples of mobile robots. Airliners, trains and cars evolve in a continuous fashion toward increasingly autonomous systems and will very probably become mobile robots in the decades to follow.

Mobile robotics is the discipline which looks at the design of mobile robots [LAU 01]. It is based on other disciplines such as automatic control, signal processing, mechanics, computing and electronics. The aim of this book is to give an overview of the tools and methods of robotics which will aid in the design of mobile robots. The robots will be modeled by *state equations*, i.e., a set of first order (mostly nonlinear) differential equations. These state equations can be obtained by using the laws of mechanics. It is not in our objectives to teach, in detail, the methods of robot modeling (refer to [JAU 05] and [JAU 15] for more information on the subject), merely to recall its principles. By *modeling*, we mean obtaining the state equations. This step is essential for simulating robots as well as designing controllers. We will however illustrate the principle of modeling in Chapter 1 on deliberately three-dimensional (3D) examples. This choice was made in order to introduce important concepts in robotics such as Euler angles and rotation matrices. For instance, we will be looking at the dynamics of a wheel and the kinematics of an underwater robot. Mobile robots are strongly nonlinear systems and only a nonlinear approach allows the construction of efficient controllers. This construction is the subject of Chapters 2 and 3. Chapter 2 is mainly based on control methods that rely on the utilization of the robot model. This approach will make use of the concept of *feedback linearization* which will be introduced and illustrated through numerous examples. Chapter 3 presents more pragmatic methods which do not use the state model of the robot and which will be referred to as *without model* or *mimetic*. The approach uses a more intuitive representation of the robot and is adapted to situations in which the robots are relatively simple to remotely control, such as in the case of cars, sailing boats or airplanes. Chapter 4 looks at *guidance*, which is placed at a higher level than control. In other words, it focuses on guiding and supervising the system which is already under control by the tools presented in Chapters 2 and 3. Therefore there will be an emphasis on finding the instruction to give to the controller in order for the robot to accomplish its given task. The guidance will then have to take into account the knowledge of the surroundings, the presence of obstacles and the roundness of the Earth. The nonlinear control and guidance methods require good knowledge of the state variables of the system, such as those which define the position of the robot. These position variables are the most difficult to find and Chapter 5 focuses on the problem of *positioning*. It introduces the classical nonlinear approaches that have been used for a very long time by humans for positioning, such as observing beacons, stars, using a compass or counting steps. Although positioning can be viewed as a particular case of state observation, the specific methods derived from it warrant a separate chapter. Chapter 6 on *identification* focuses on finding, with a certain precision, non-measured quantities (parameters and position) from other, measured quantities. In order to perform this identification, we will mainly be looking at the so-called *least squares* approach which consists of finding the vector of variables that minimizes the sum of the squares of the errors. Chapter 7 presents the *Kalman filter*. This filter can be seen as a state observer for dynamic linear systems with coefficients that vary in time.

The MATLAB code related to the exercises of this book together with explanatory videos can be found at the following address:

`www.ensta-bretagne.fr/jaulin/isterob.html`

Contents

Introduction ix

Chapter 1. Three-dimensional Modeling 1

1.1. Rotation matrices 2
1.1.1. Definition 2
1.1.2. Rotation vector 3
1.1.3. Adjoint 4
1.1.4. Coordinate system change 6
1.2. Euler angles 8
1.2.1. Definition 8
1.2.2. Derivative of an Euler matrix 9
1.2.3. Rotation vector of an Euler matrix 10
1.3. Kinematic model of a solid robot 11
1.4. Exercises 14
1.5. Corrections 28

Chapter 2. Feedback Linearization 45

2.1. Controlling an integrator chain 45
2.1.1. Proportional-derivative controller 46
2.1.2. Proportional-integral-derivative controller 47
2.2. Introductory example 48
2.3. Principle of the method 50
2.3.1. Principle 50
2.3.2. Relative degree 52
2.3.3. Differential delay matrix 53
2.3.4. Singularities 54
2.4. Cart 56

2.4.1. First model 56
2.4.2. Second model 58
2.5. Controlling a tricycle 61
2.5.1. Speed and heading control 61
2.5.2. Position control 63
2.5.3. Choosing another output 63
2.6. Sailboat 64
2.6.1. Polar curve 66
2.6.2. Differential delay 67
2.6.3. The method of feedback linearization 67
2.6.4. Polar curve control 70
2.7. Kinematic model and dynamic model 71
2.7.1. Principle 71
2.7.2. Example of the inverted rod pendulum 73
2.7.3. Servo-motors 76
2.8. Exercises 77
2.9. Corrections 85

CHAPTER 3. MODEL-FREE CONTROL 101

3.1. Model-free control of a robot cart 102
3.1.1. Proportional heading and speed controller 102
3.1.2. Proportional-derivative heading controller 104
3.2. Skate car 106
3.2.1. Model 106
3.2.2. Sinusoidal control 108
3.2.3. Maximum thrust control 109
3.2.4. Simplification of the fast dynamics 111
3.3. Sailboat 114
3.3.1. Problem 114
3.3.2. Controller 116
3.3.3. Navigation 121
3.3.4. Experiment 122
3.4. Exercises 125
3.5. Corrections 133

CHAPTER 4. GUIDANCE 141

4.1. Guidance on a sphere 141
4.2. Path planning 145
4.2.1. Simple example 145
4.2.2. Bézier polynomials 146
4.3. Voronoi diagram 148

4.4. Artificial potential field method 149
4.5. Exercises 151
4.6. Corrections 157

CHAPTER 5. INSTANTANEOUS LOCALIZATION 171

5.1. Sensors 172
5.2. Goniometric localization 176
5.2.1. Formulation of the problem 176
5.2.2. Inscribed angles 177
5.2.3. Static triangulation of a plane robot 178
5.2.4. Dynamic triangulation 179
5.3. Multilateration 181
5.4. Exercises 182
5.5. Corrections 187

CHAPTER 6. IDENTIFICATION 197

6.1. Quadratic functions 197
6.1.1. Definition 198
6.1.2. Derivative of a quadratic form 198
6.1.3. Eigenvalues of a quadratic function 199
6.1.4. Minimizing a quadratic function 199
6.2. The least squares method 201
6.2.1. Linear case 201
6.2.2. Nonlinear case 202
6.3. Exercises 203
6.4. Corrections 207

CHAPTER 7. KALMAN FILTER 219

7.1. Covariance matrices 219
7.1.1. Definitions and interpretations 219
7.1.2. Properties 221
7.1.3. Confidence ellipse 222
7.1.4. Generating Gaussian random vectors 224
7.2. Unbiased orthogonal estimator 226
7.3. Application to linear estimation 230
7.4. Kalman filter 231
7.5. Kalman smoother 234
7.6. Exercises 235
7.7. Corrections 256

BIBLIOGRAPHY 295

INDEX 299

1

Three-dimensional Modeling

This chapter presents the three-dimensional (3D) modeling of a solid (non-articulated) robot. Such a modeling is used to represent an airplane, a quadcopter, a submarine and so forth. Through this modeling, we will introduce a number of fundamental concepts in robotics such as state representation, rotation matrices and Euler angles. The robots, whether mobile, manipulator or articulated, can generally be put into a state representation form:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)) \end{cases}$$

where $\mathbf{x}$ is the state vector, $\mathbf{u}$ is the input vector and $\mathbf{y}$ is the vector of measurements [JAU 05]. We will call *modeling* the step which consists of finding a more or less accurate state representation of the robot in question. In general, constant parameters may appear in the state equations (such as the mass and moment of inertia of a body, viscosity, etc.). In such cases, an identification step might prove to be necessary. We will assume that all of the parameters are known. Of course, there is no systematic methodology that can be applied for modeling a mobile robot. The aim of this chapter is to present the tools which allow us to reach a state representation of 3D solid robots in order for the readers to acquire a certain experience which will be helpful when modeling his/her own robots. This modeling will also allow us to recall a number of important concepts in Euclidean geometry, which are fundamental in mobile robotics. This chapter begins by recalling a number of important concepts in kinematics which will be useful for the modeling.

1.1. Rotation matrices

For 3D modeling, it is essential to have a good understanding of the concepts related to rotation matrices, which are recalled in this section. It is by using this tool that we will perform our coordinate system transformations and position our objects in space.

1.1.1. *Definition*

Let us recall that the j^{th} column of the matrix of a linear application of $\mathbb{R}^n \to \mathbb{R}^n$ represents the image of the j^{th} vector $\mathbf{e}_j$ of the standard basis (see Figure 1.1). Thus, the expression of a rotation matrix of angle θ in the plane $\mathbb{R}^2$ is given by:

$$\mathbf{R} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

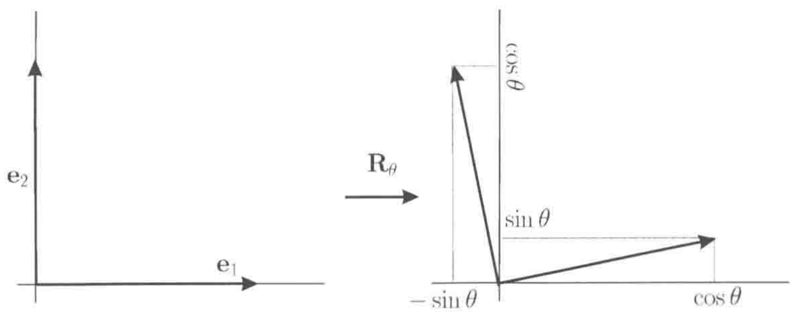

Figure 1.1. *Rotation of angle θ in a plane*

Concerning rotations in the space $\mathbb{R}^3$ (see Figure 1.2), it is important to specify the axis of rotation. We distinguish three main rotations: the rotation around the Ox axis, the rotation around the Oy axis and the rotation around the Oz axis.

The associated matrices are, respectively, given by:

$$\mathbf{R}_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{pmatrix}, \mathbf{R}_y = \begin{pmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{pmatrix},$$

$$\mathbf{R}_z = \begin{pmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Let us recall the formal definition of a rotation. A rotation is a linear application which is an isometry (in other words, it preserves the scalar product) and a movement (it does not change the orientation in space).

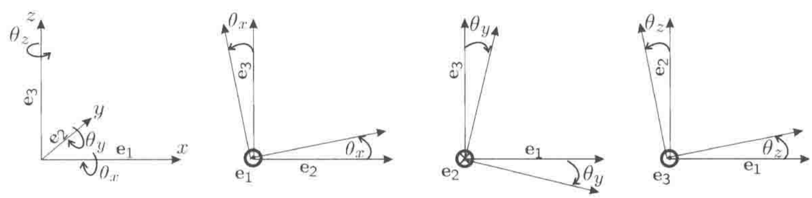

Figure 1.2. *Rotations in $\mathbb{R}^3$ following various viewing angles*

THEOREM 1.1.– A matrix $\mathbf{R}$ is a rotation matrix if and only if:

$$\mathbf{R}^{\mathrm{T}} \cdot \mathbf{R} = \mathbf{I} \text{ and } \det \mathbf{R} = 1$$

PROOF.– The scalar product is preserved by $\mathbf{R}$ if, for any $\mathbf{u}$ and $\mathbf{v}$ in $\mathbb{R}^n$, we have:

$$(\mathbf{Ru})^{\mathrm{T}} \cdot (\mathbf{Rv}) = \mathbf{u}^{\mathrm{T}}\mathbf{R}^{\mathrm{T}}\mathbf{R}\mathbf{v} = \mathbf{u}^{\mathrm{T}}\mathbf{v}$$

in other words, if $\mathbf{R}^{\mathrm{T}}\mathbf{R} = \mathbf{I}$. The symmetries relative to a plane, as well as all the other improper movements (isometries that change the orientation of space, such as a mirror), also verify the property $\mathbf{R}^{\mathrm{T}} \cdot \mathbf{R} = \mathbf{I}$. The condition $\det \mathbf{R} = 1$ allows us to be limited to the isometries which are movements . The set of rotation matrices of $\mathbb{R}^n$ forms a group referred to as a *special orthogonal group* (special because $\det \mathbf{R} = 1$, orthogonal because $\mathbf{R}^{\mathrm{T}} \cdot \mathbf{R} = \mathbf{I}$). ■

1.1.2. *Rotation vector*

If $\mathbf{R}$ is a rotation matrix depending on time t, by differentiating the relation $\mathbf{R}\mathbf{R}^{\mathrm{T}} = \mathbf{I}$, we obtain:

$$\dot{\mathbf{R}} \cdot \mathbf{R}^{\mathrm{T}} + \mathbf{R} \cdot \dot{\mathbf{R}}^{\mathrm{T}} = \mathbf{0}$$

Thus, the matrix $\dot{\mathbf{R}} \cdot \mathbf{R}^{\mathrm{T}}$ is a skew-symmetric matrix (in other words, it satisfies $\mathbf{A}^{\mathrm{T}} = -\mathbf{A}$ and therefore its diagonal contains only zeroes, and for each element of $\mathbf{A}$, we have $a_{ij} = -a_{ji}$). Therefore, we may write, in the case where $\mathbf{R}$ is of dimension 3×3:

$$\dot{\mathbf{R}} \cdot \mathbf{R}^{\mathrm{T}} = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix} \qquad [1.1]$$

The vector $\omega = (\omega_x, \omega_y, \omega_z)$ is called the *rotation vector* associated with the pair $\left(\mathbf{R}, \dot{\mathbf{R}}\right)$. It must be noted that $\dot{\mathbf{R}}$ is not a matrix with good properties (for instance the fact of being skew-symmetric). However, the matrix $\dot{\mathbf{R}} \cdot \mathbf{R}^{\mathrm{T}}$ has the [1.1] structure since it allows us to be positioned within the coordinate system in which the rotation is performed and this, due to the change of basis performed by $\mathbf{R}^{\mathrm{T}}$. We will define the *vector product* between two vectors ω and $\mathbf{x} \in \mathbb{R}^3$ as follows:

$$\omega \wedge \mathbf{x} = \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} \wedge \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} x_3\omega_y - x_2\omega_z \\ x_1\omega_z - x_3\omega_x \\ x_2\omega_x - x_1\omega_y \end{pmatrix}$$

$$= \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

1.1.3. *Adjoint*

For each vector $\omega = (\omega_x, \omega_y, \omega_z)$, we may adjoin the skew-symmetric matrix:

$$\mathbf{Ad}\,(\omega) \stackrel{\text{def}}{=} \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix}$$

which can be interpreted as the matrix associated with a vector product by the vector ω.