Chapter 5

Direct kinematic model of serial robots

5.1. Introduction

The direct kinematic model of a robot manipulator gives the velocity of the end-effector $\dot{\mathbf{X}}$ in terms of the joint velocities $\dot{\mathbf{q}}$. It is written as:

$$\dot{\mathbf{X}} = \mathbf{J}(\mathbf{q})\,\dot{\mathbf{q}} \qquad [5.1]$$

where $\mathbf{J}(\mathbf{q})$ denotes the (mxn) Jacobian matrix.

The same Jacobian matrix also appears in the direct differential model, which provides the differential displacement of the end-effector $\mathbf{dX}$ in terms of the differential variation of the joint variables $\mathbf{dq}$:

$$\mathbf{dX} = \mathbf{J}(\mathbf{q})\,\mathbf{dq} \qquad [5.2]$$

The Jacobian matrix has multiple applications in robotics [Whitney 69], [Paul 81]. The most obvious is the use of its inverse to numerically compute a solution for the inverse geometric model, i.e. to compute the joint variables $\mathbf{q}$ corresponding to a given location of the end-effector $\mathbf{X}$ (Chapter 6). The transpose of the Jacobian matrix is used in the static model to compute the necessary joint forces and torques to exert specified forces and moments on the environment. The Jacobian matrix is also used to determine the singularities and to analyze the reachable workspace of robots [Borrel 86], [Wenger 89].

In this chapter, we will present the computation of the Jacobian matrix and expose its different applications for serial robots. The kinematic model of complex chain robots will be studied in Chapter 7.

5.2. Computation of the Jacobian matrix from the direct geometric model

The Jacobian matrix can be obtained by differentiating the DGM, $\mathbf{X} = \mathbf{f}(\mathbf{q})$, using the partial derivative $\frac{\partial \mathbf{f}}{\partial \mathbf{q}}$ such that:

$$J_{ij} = \frac{\partial f_i(\mathbf{q})}{\partial q_j} \quad \text{for } i = 1, \ldots, m \text{ and } j = 1, \ldots, n \qquad [5.3]$$

where J_{ij} is the (i, j) element of the Jacobian matrix **J**.

This method is convenient for simple robots having a reduced number of degrees of freedom as shown in the following example. The computation of the basic Jacobian matrix, also known as kinematic Jacobian matrix, is more practical for a general n degree-of-freedom robot. It is presented in § 5.3.

• **Example 5.1.** Let us consider the three degree-of-freedom planar robot presented in Figure 5.1. Let us denote the link lengths by L1, L2 and L3.

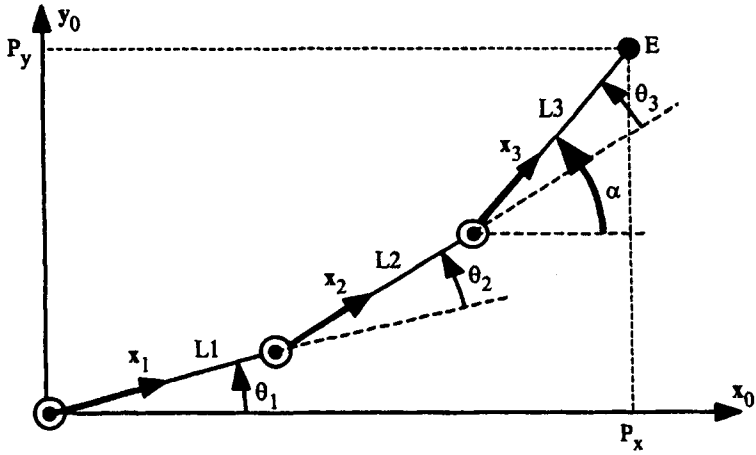

Figure 5.1. *Example of a three degree-of-freedom planar robot*

The task coordinates, defined as the position coordinates (P_x, P_y) of the terminal point E and the angle α giving the orientation of the third link relative to frame R_0, are such that:

$$P_x = C1\ L1 + C12\ L2 + C123\ L3$$
$$P_y = S1\ L1 + S12\ L2 + S123\ L3$$

$$\alpha = \theta_1 + \theta_2 + \theta_3$$

where C1 = $\cos(\theta_1)$, S1 = $\sin(\theta_1)$, C12 = $\cos(\theta_1+\theta_2)$, S12 = $\sin(\theta_1+\theta_2)$, C123 = $\cos(\theta_1+\theta_2+\theta_3)$ and S123 = $\sin(\theta_1+\theta_2+\theta_3)$.

The Jacobian matrix is obtained by differentiating these expressions with respect to θ_1, θ_2 and θ_3:

$$\mathbf{J} = \begin{bmatrix} -S1L1-S12L2-S123L3 & -S12L2-S123L3 & -S123L3 \\ C1L1+C12L2+C123L3 & C12L2+C123L3 & C123L3 \\ 1 & 1 & 1 \end{bmatrix}$$

5.3. Basic Jacobian matrix

In this section, we present a direct method to compute the Jacobian matrix of a serial mechanism without differentiating the DGM. The Jacobian matrix obtained is called the *basic Jacobian matrix*, or *kinematic Jacobian matrix*. It relates the kinematic screw of frame R_n to the joint velocities $\dot{\mathbf{q}}$:

$$\mathbb{V}_n = \begin{bmatrix} \mathbf{V}_n \\ \boldsymbol{\omega}_n \end{bmatrix} = \mathbf{J}_n \dot{\mathbf{q}} \qquad [5.4a]$$

where $\mathbf{V}_n$ and $\boldsymbol{\omega}_n$ are the linear and angular velocities of frame R_n respectively. We note that $\mathbf{V}_n$ is the derivative of the position vector $\mathbf{P}_n$ with respect to time, while $\boldsymbol{\omega}_n$ is not the derivative of any orientation vector.

The basic Jacobian matrix also gives the relationship between the differential translation and rotation vectors ($\mathbf{dP}_n$, $\boldsymbol{\delta}_n$) of frame R_n in terms of the differential joint variables $\mathbf{dq}$:

$$\begin{bmatrix} \mathbf{dP}_n \\ \boldsymbol{\delta}_n \end{bmatrix} = \mathbf{J}_n \, \mathbf{dq} \qquad [5.4b]$$

We will show in § 5.11 that the Jacobian giving the end-effector velocity $\dot{\mathbf{X}}$, for any task coordinate representation, can be deduced from the basic Jacobian $\mathbf{J}_n$.

5.3.1. *Computation of the basic Jacobian matrix*

The velocity $\dot{q}_k$ of joint k produces the linear and angular velocities ($\mathbf{V}_{k,n}$ and $\omega_{k,n}$ respectively) at the terminal frame R_n. Two cases are considered:

- if joint k is prismatic ($\sigma_k = 1$, Figure 5.2):

$$\begin{cases} \mathbf{V}_{k,n} = \mathbf{a}_k \dot{q}_k \\ \omega_{k,n} = \mathbf{0} \end{cases} \qquad [5.5]$$

  where $\mathbf{a}_k$ is the unit vector along the $\mathbf{z}_k$ axis;

- if joint k is revolute ($\sigma_k = 0$, Figure 5.3):

$$\begin{cases} \mathbf{V}_{k,n} = \mathbf{a}_k \dot{q}_k \times \mathbf{L}_{k,n} = (\mathbf{a}_k \times \mathbf{L}_{k,n}) \dot{q}_k \\ \omega_{k,n} = \mathbf{a}_k \dot{q}_k \end{cases} \qquad [5.6]$$

  where $\mathbf{L}_{k,n}$ denotes the position vector connecting O_k to O_n.

Thus, $\mathbf{V}_{k,n}$ and $\omega_{k,n}$ can be written in the following general form:

$$\begin{cases} \mathbf{V}_{k,n} = [\sigma_k \mathbf{a}_k + \bar{\sigma}_k (\mathbf{a}_k \times \mathbf{L}_{k,n})] \dot{q}_k \\ \omega_{k,n} = \bar{\sigma}_k \mathbf{a}_k \dot{q}_k \end{cases} \qquad [5.7]$$

The linear and angular velocities of the terminal frame can be written as:

$$\begin{cases} \mathbf{V}_n = \sum_{k=1}^{n} \mathbf{V}_{k,n} = \sum_{k=1}^{n} [\sigma_k \mathbf{a}_k + \bar{\sigma}_k (\mathbf{a}_k \times \mathbf{L}_{k,n})] \dot{q}_k \\ \omega_n = \sum_{k=1}^{n} \omega_{k,n} = \sum_{k=1}^{n} \bar{\sigma}_k \mathbf{a}_k \dot{q}_k \end{cases} \qquad [5.8]$$

Writing equation [5.8] in matrix form and using equation [5.4], we deduce that:

$$\mathbf{J}_n = \begin{bmatrix} \sigma_1 \mathbf{a}_1 + \bar{\sigma}_1 (\mathbf{a}_1 \times \mathbf{L}_{1,n}) & \cdots & \sigma_n \mathbf{a}_n + \bar{\sigma}_n (\mathbf{a}_n \times \mathbf{L}_{n,n}) \\ \bar{\sigma}_1 \mathbf{a}_1 & \cdots & \bar{\sigma}_n \mathbf{a}_n \end{bmatrix} \qquad [5.9]$$

Referring the vectors of $\mathbf{J}_n$ with respect to frame R_i, we obtain the (6xn) Jacobian matrix ${}^i\mathbf{J}_n$ such that:

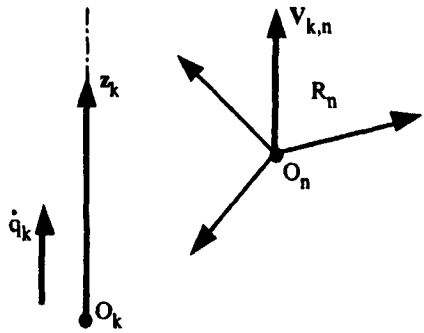

Figure 5.2. *Case of a prismatic joint*

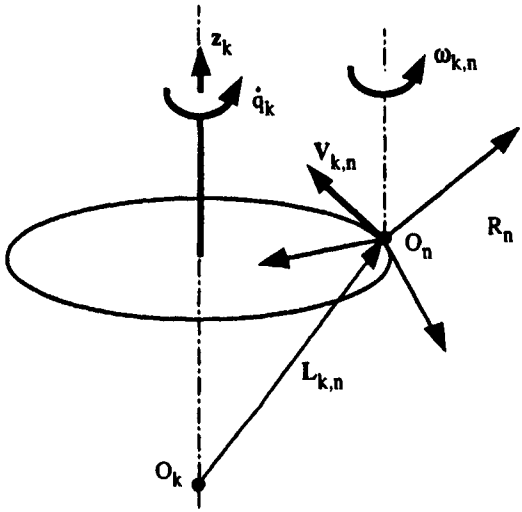

Figure 5.3. *Case of a revolute joint*

$$^{i}\mathbb{V}_n = {}^{i}J_n\,\dot{q} \qquad [5.10]$$

In general, we calculate V_n and ω_n in frame R_n or frame R_0. The corresponding Jacobian matrix is denoted by $^{n}J_n$ or $^{0}J_n$ respectively. These matrices can also be computed using any matrix $^{i}J_n$, for i = 0, ..., n, thanks to the following expression:

$$^{s}J_n = \begin{bmatrix} {}^{s}A_i & 0_3 \\ 0_3 & {}^{s}A_i \end{bmatrix} {}^{i}J_n \qquad [5.11]$$

where ${}^{s}\mathbf{A}_i$ is the (3x3) orientation matrix of frame R_i relative to frame R_s.

In general, we obtain the simplest matrix ${}^{i}\mathbf{J}_n$ when i = integer (n/2). We note that the matrices ${}^{i}\mathbf{J}_n$, for i = 0, ..., n, have the same singular positions.

5.3.2. *Computation of the matrix ${}^{i}J_n$*

Since the vector product $\mathbf{a}_k \times \mathbf{L}_{k,n}$ can be computed by $\hat{\mathbf{a}}_k \mathbf{L}_{k,n}$, the k^{th} column of ${}^{i}\mathbf{J}_n$, denoted as ${}^{i}\mathbf{j}_{n;k}$, becomes:

$$ {}^{i}\mathbf{j}_{n;k} = \begin{bmatrix} \sigma_k \, {}^{i}\mathbf{a}_k + \bar{\sigma}_k \, {}^{i}\mathbf{A}_k \, {}^{k}\hat{\mathbf{a}}_k \, {}^{k}\mathbf{L}_{k,n} \\ \\ \bar{\sigma}_k \, {}^{i}\mathbf{a}_k \end{bmatrix} $$

Since ${}^{k}\mathbf{a}_k = [0 \quad 0 \quad 1]^T$ and ${}^{k}\mathbf{L}_{k,n} = {}^{k}\mathbf{P}_n$, we obtain:

$$ {}^{i}\mathbf{j}_{n;k} = \begin{bmatrix} \sigma_k \, {}^{i}\mathbf{a}_k + \bar{\sigma}_k \, (-{}^{k}P_{ny} \, {}^{i}\mathbf{s}_k + {}^{k}P_{nx} \, {}^{i}\mathbf{n}_k) \\ \\ \bar{\sigma}_k \, {}^{i}\mathbf{a}_k \end{bmatrix} \qquad [5.12] $$

where ${}^{k}P_{nx}$ and ${}^{k}P_{ny}$ denote the x and y components of the vector ${}^{k}\mathbf{P}_n$ respectively.

From this expression, we obtain the k^{th} column of ${}^{n}\mathbf{J}_n$ as:

$$ {}^{n}\mathbf{j}_{n;k} = \begin{bmatrix} \sigma_k \, {}^{n}\mathbf{a}_k + \bar{\sigma}_k \, (-{}^{k}P_{ny} \, {}^{n}\mathbf{s}_k + {}^{k}P_{nx} \, {}^{n}\mathbf{n}_k) \\ \\ \bar{\sigma}_k \, {}^{n}\mathbf{a}_k \end{bmatrix} \qquad [5.13] $$

The column ${}^{n}\mathbf{j}_{n;k}$ is computed from the elements of the matrix ${}^{k}\mathbf{T}_n$ resulting from the DGM.

In a similar way, the k^{th} column of ${}^{i}\mathbf{J}_n$ is also written as:

$$ {}^{i}\mathbf{j}_{n;k} = \begin{bmatrix} \sigma_k \, {}^{i}\mathbf{a}_k + \bar{\sigma}_k \, {}^{i}\hat{\mathbf{a}}_k \, ({}^{i}\mathbf{P}_n - {}^{i}\mathbf{P}_k) \\ \\ \bar{\sigma}_k \, {}^{i}\mathbf{a}_k \end{bmatrix} \qquad [5.14] $$

which gives for i = 0:

$$ {}^{0}\mathbf{J}_{n;k} = \begin{bmatrix} \sigma_k \, {}^{0}\mathbf{a}_k + \bar{\sigma}_k \, {}^{0}\hat{\mathbf{a}}_k \, ({}^{0}\mathbf{P}_n - {}^{0}\mathbf{P}_k) \\ \bar{\sigma}_k \, {}^{0}\mathbf{a}_k \end{bmatrix} \qquad [5.15] $$

In this case, we need to compute the matrices ${}^{0}\mathbf{T}_k$ for k = 1, ..., n.

NOTE. – To find the Jacobian ${}^{E}\mathbf{J}_E$ defining the velocity of the tool frame R_E, we can either make use of equation [5.9] after replacing $\mathbf{L}_{k,n}$ by $\mathbf{L}_{k,E}$, or compute ${}^{E}\mathbb{V}_E$ as a function of ${}^{n}\mathbb{V}_n$, and deduce ${}^{E}\mathbf{J}_E$. From § 2.4.3, we can see that:

$$ {}^{E}\mathbb{V}_E = {}^{E}\mathbb{T}_n \, {}^{n}\mathbb{V}_n = {}^{E}\mathbb{T}_n \, {}^{n}\mathbf{J}_n \, \dot{\mathbf{q}} $$

where ${}^{E}\mathbb{T}_n$ is the (6x6) screw transformation matrix defined in equation [2.47]. Consequently, we deduce that:

$$ {}^{E}\mathbf{J}_E = \begin{bmatrix} {}^{E}\mathbf{A}_n & -{}^{E}\mathbf{A}_n \, {}^{n}\hat{\mathbf{P}}_E \\ \mathbf{0}_3 & {}^{E}\mathbf{P}_n \end{bmatrix} {}^{n}\mathbf{J}_n = {}^{E}\mathbb{T}_n \, {}^{n}\mathbf{J}_n \qquad [5.16] $$

• **Example 5.2.** Compute the Jacobian matrix ${}^{6}\mathbf{J}_6$ of the Stäubli RX-90 robot. Using equation [5.13] and the matrices ${}^{k}\mathbf{T}_6$ resulting from the DGM, we obtain:

J (1,1) = (– C6C5S4 – S6C4)(S23RL4 – C2D3)
J (2,1) = (S6C5S4 – C6C4)(S23RL4 – C2D3)
J (3,1) = S5S4(S23RL4 – C2D3)
J (4,1) = (C6C5C4 – S6S4)S23 + C6S5C23
J (5,1) = (– S6C5C4 – C6S4)S23 – S6S5C23
J (6,1) = – S5C4S23 + C5C23
J (1,2) = (– C6C5C4 + S6S4)(RL4 – S3D3) + C6S5C3D3
J (2,2) = (S6C5C4 + C6S4)(RL4 – S3D3) – S6S5C3D3
J (3,2) = S5C4(RL4 – S3D3) + C5C3D3
J (4,2) = – C6C5S4 – S6C4
J (5,2) = S6C5S4 – C6C4
J (6,2) = S5S4
J (1,3) = (– C6C5C4 + S6S4)RL4
J (2,3) = (S6C5C4 + C6S4)RL4
J (3,3) = S5C4RL4
J (4,3) = – C6C5S4 – S6C4
J (5,3) = S6C5S4 – C6C4
J (6,3) = S5S4
J (1,4) = 0
J (2,4) = 0
J (3,4) = 0
J (4,4) = C6S5
J (5,4) = – S6S5

J (6,4) = C5
J (1,5) = 0
J (2,5) = 0
J (3,5) = 0
J (4,5) = – S6
J (5,5) = – C6
J (6,5) = 0
J (1,6) = 0
J (2,6) = 0
J (3,6) = 0
J (4,6) = 0
J (5,6) = 0
J (6,6) = 1

• **Example 5.3.** Determine the Jacobian matrix ${}^3\mathbf{J}_6$ of the Stäubli RX-90 robot. The column k of the matrix ${}^3\mathbf{J}_6$ for a revolute joint is obtained from equation [5.12] as:

$$ {}^3\mathbf{j}_{6;k} = \begin{bmatrix} -{}^k\mathbf{P}_{6y}\,{}^3\mathbf{s}_k + {}^k\mathbf{P}_{6x}\,{}^3\mathbf{n}_k \\ {}^3\mathbf{a}_k \end{bmatrix} $$

The elements ${}^k\mathbf{P}_{6y}$ and ${}^k\mathbf{P}_{6x}$ are obtained from the DGM. The vectors ${}^3\mathbf{s}_k$, ${}^3\mathbf{n}_k$ and ${}^3\mathbf{a}_k$, for k = 2, 3, 4 and 6, are deduced from the matrices ${}^3\mathbf{A}_2$, ${}^3\mathbf{A}_3$, ${}^3\mathbf{A}_4$ and ${}^3\mathbf{A}_6$, which are also computed for the DGM. The additional matrices to be computed are ${}^3\mathbf{A}_1$ and ${}^3\mathbf{A}_5$. Finally, we obtain:

$$ {}^3\mathbf{J}_6 = \begin{bmatrix} 0 & -\text{RL4}+\text{S3D3} & -\text{RL4} & 0 & 0 & 0 \\ 0 & \text{C3D3} & 0 & 0 & 0 & 0 \\ \text{S23 RL4}-\text{C2D3} & 0 & 0 & 0 & 0 & 0 \\ \text{S23} & 0 & 0 & 0 & \text{S4} & -\text{S5C4} \\ \text{C23} & 0 & 0 & 1 & 0 & \text{C5} \\ 0 & 1 & 1 & 0 & \text{C4} & \text{S5S4} \end{bmatrix} $$

5.4. Decomposition of the Jacobian matrix into three matrices

We have shown in equation [5.11] that the matrix ${}^s\mathbf{J}_n$ could be decomposed into two matrices; the first is always of full-rank and the second contains simple elements. Renaud [Renaud 80b] has shown that the Jacobian matrix could also be decomposed into three matrices: the first two are always of full-rank and their inverse is straightforward; the third is of the same rank as ${}^s\mathbf{J}_n$, but contains simpler elements.

Figure 5.4 illustrates the principle of the proposed method: the influence of the joint velocities is not calculated at the level of the terminal frame R_n but at the level of an intermediate frame R_j. Therefore, we define the Jacobian matrix $J_{n,j}$ as:

$$J_{n,j} = \begin{bmatrix} \sigma_1 a_1 + \bar{\sigma}_1 (a_1 \times L_{1,j}) & \dots & \sigma_n a_n + \bar{\sigma}_n (a_n \times L_{n,j}) \\ \bar{\sigma}_1 a_1 & \dots & \bar{\sigma}_n a_n \end{bmatrix} \quad [5.17]$$

The matrix $J_{n,j}$ is equivalent to the Jacobian matrix defining the velocity of a frame fixed to link n and aligned instantaneously with frame R_j. We can compute J_n from $J_{n,j}$ using the expression:

$$J_n = \begin{bmatrix} I_3 & -\hat{L}_{j,n} \\ 0_3 & I_3 \end{bmatrix} J_{n,j} \quad [5.18]$$

By projecting the elements of this equation into frame R_i, we obtain:

$${}^{i}J_n = \begin{bmatrix} I_3 & -{}^{i}\hat{L}_{j,n} \\ 0_3 & I_3 \end{bmatrix} {}^{i}J_{n,j} \quad [5.19]$$

with:

$${}^{i}L_{j,n} = {}^{i}A_j \, {}^{j}P_n \quad [5.20]$$

The k^{th} column of ${}^{i}J_{n,j}$, deduced from equation [5.17], is expressed in frame R_i as:

$${}^{i}J_{n,j;k} = \begin{bmatrix} \sigma_k \, {}^{i}a_k + \bar{\sigma}_k \, (-{}^{k}P_{jy} \, {}^{i}s_k + {}^{k}P_{jx} \, {}^{i}n_k) \\ \bar{\sigma}_k \, {}^{i}a_k \end{bmatrix} \quad [5.21]$$

We note that ${}^{i}J_n = {}^{i}J_{n,n}$. Thus, the matrix ${}^{s}J_n$ can be expressed by the multiplication of the following three matrices where the first two are of full-rank:

$${}^{s}J_n = \begin{bmatrix} {}^{s}A_i & 0_3 \\ 0_3 & {}^{s}A_i \end{bmatrix} \begin{bmatrix} I_3 & -{}^{i}\hat{L}_{j,n} \\ 0_3 & I_3 \end{bmatrix} {}^{i}J_{n,j} \quad [5.22]$$

In general, the shift frame R_j and the projection frame R_i leading to a simple matrix ${}^{i}J_{n,j}$ are chosen such that i = integer (n/2) and j = i + 1.

Thus, for a six degree-of-freedom robot, the simplest Jacobian matrix is ${}^3\mathbf{J}_{6,4}$. If the robot has a spherical wrist, the vector $\mathbf{L}_{4,6}$ is zero and consequently ${}^3\mathbf{J}_{6,4} = {}^3\mathbf{J}_6$.

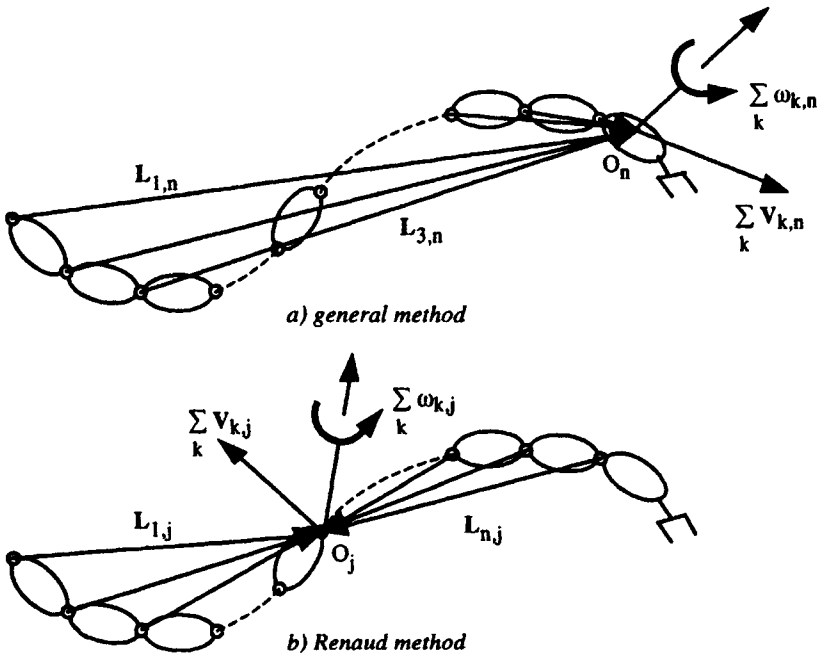

Figure 5.4. *Principle of Renaud method*

5.5. Efficient computation of the end-effector velocity

Having calculated $\mathbf{J}_n$, the linear and angular velocities $\mathbf{V}_n$ and ω_n of frame $\mathbf{R}_n$ can be obtained from equation [5.4a]. However, in order to reduce the computational cost, it is more efficient, as we will see in Chapter 9, to use the following recursive equations for j = 1, ..., n:

$$\begin{cases} {}^j\omega_{j-1} = {}^jA_{j-1}\,{}^{j-1}\omega_{j-1} \\ {}^j\omega_j = {}^j\omega_{j-1} + \bar{\sigma}_j\,\dot{q}_j\,{}^j\mathbf{a}_j \\ {}^jV_j = {}^jA_{j-1}\,({}^{j-1}V_{j-1} + {}^{j-1}\omega_{j-1} \times {}^{j-1}P_j) + \sigma_j\,\dot{q}_j\,{}^j\mathbf{a}_j \end{cases} \qquad [5.23]$$

where ${}^j\mathbf{a}_j$ is the unit vector $[0\ \ 0\ \ 1]^T$. We initialize the algorithm by the linear and angular velocities of the robot base ($\mathbf{V}_0$ and ω_0 respectively), which are zero if the base is fixed.

5.6. Dimension of the task space of a robot

At a given joint configuration $\mathbf{q}$, the rank r of the Jacobian matrix ${}^{i}\mathbf{J}_{n}$ – hereafter written as $\mathbf{J}$ to simplify the notation – corresponds to the number of degrees of freedom of the end-effector. It defines the dimension of the accessible task space at this configuration. The number of degrees of freedom of the task space of a robot, M, is equal to the maximum rank, r_{max}, which the Jacobian matrix can have at all possible configurations. Noting the number of degrees of freedom of the robot as N (equal to n for serial robots), the following cases are considered [Gorla 84]:

- if $N = M$, the robot is non-redundant and has just the number of joints required to provide M degrees of freedom to the end-effector;
- if $N > M$, the robot is redundant of order $(N - M)$. It has more joints than required to provide M degrees of freedom to the end-effector;
- if $r < M$, the Jacobian matrix is rank deficient. The robot is at a singular configuration of order $(M - r)$. At this configuration, the robot cannot generate an end-effector velocity along some directions of the task space, which are known as *degenerate directions*. When the matrix $\mathbf{J}$ is square, the singularities are obtained by the zeros of $\det(\mathbf{J}) = 0$, where $\det(\mathbf{J})$ indicates the determinant of the Jacobian matrix of the robot. They correspond to the zeros of $\det(\mathbf{J}\,\mathbf{J}^{T}) = 0$ for redundant robots.

• **Example 5.4.** Computation of the singularities of the Stäubli RX-90 robot. Noting that the Jacobian matrix ${}^{3}\mathbf{J}_{6}$ (obtained in Example 5.3) has the following particular form:

$$ {}^{3}\mathbf{J}_{6} = \begin{bmatrix} \mathbf{A} & \mathbf{0}_{3} \\ \mathbf{B} & \mathbf{C} \end{bmatrix} $$

we obtain $\det({}^{3}\mathbf{J}_{6}) = \det(\mathbf{A}) \det(\mathbf{C}) = -C3\ D3\ RL4\ S5\ (S23\ RL4 - C2\ D3)$.

The maximum rank is $r_{max} = 6$. The robot is not redundant because it has six degrees of freedom. However, this rank drops to five in the following three singular configurations:

$$ \begin{cases} C3 = 0 \\ S23\ RL4 - C2\ D3 = 0 \\ S5 = 0 \end{cases} $$

- when $C3 = 0$, the robot is fully extended (Figure 4.2c) or fully folded. In this case, the origin O_6 is located on the boundary of its workspace: this elbow singularity has not been deduced from the inverse geometric model (§ 4.3.2,

Example 4.1). In this configuration, where the second row of ${}^3\mathbf{J}_6$ is zero, the robot cannot generate linear velocity for O_6 along the direction O_6O_2;

- the singularity S23 RL4 – C2 D3 = 0 (Figure 4.2a), already deduced from the inverse geometric model, corresponds to a configuration in which O_6 is located on the $\mathbf{z}_0$ axis (shoulder singularity). In this configuration, where $P_x = P_y = 0$, the third row of ${}^3\mathbf{J}_6$ is zero. The robot cannot generate velocity for O_6 along the normal to the plane containing the points O_2, O_3 and O_6;
- for S5 = 0 (Figure 4.2b), the axes of the joints θ_4 and θ_6 are aligned, resulting in the loss of one degree of freedom of the robot. We notice that the columns 4 and 6 of ${}^3\mathbf{J}_6$ are identical. In this configuration, the robot cannot generate rotational velocity for frame R_6 about the normal to the plane containing the axes $\mathbf{z}_4$, $\mathbf{z}_5$ and $\mathbf{z}_6$. This wrist singularity has already been deduced from the inverse geometric model.

5.7. Analysis of the robot workspace

The analysis of the workspace is very important for the design, selection and programming of robots.

5.7.1. *Workspace*

Let $\mathbf{q} = [q_1, \ldots, q_n]$ be an element of the joint space and let $\mathbf{X} = [x_1, \ldots, x_m]$ be the corresponding element in the task space, such that:

$$\mathbf{X} = \mathbf{f}(\mathbf{q}) \qquad [5.24]$$

The joint domain $\mathbf{Q}$ is defined as the set of all reachable configurations taking into account the joint limits:

$$\mathbf{Q} = \{\mathbf{q} \mid q_{i\min} \leq q_i \leq q_{i\max}, \forall\, i = 1, \ldots, n\} \qquad [5.25]$$

The image of $\mathbf{Q}$ by the direct geometric model DGM defines the workspace W of the robot:

$$\mathbf{W} = \mathbf{f}(\mathbf{Q}) \qquad [5.26]$$

Thus, the workspace W is the set of positions and orientations reachable by the robot end-effector. Its geometry depends on the robot architecture. Its boundaries are defined by the singularities and the joint limits. However, when there is an obstacle in the robot workspace, additional boundaries limiting the reachable zones will appear [Wenger 89].

For robots with two joints, the workspace is easy to obtain and can be visualized in a plane. For a three degree-of-freedom positioning shoulder, the workspace can be represented by a generic planar cross section of W. This cross section contains the axis of the first joint if it is revolute, whereas it is perpendicular to the axis of the first joint if it is prismatic. The whole workspace is obtained from the generic cross section by rotating it about (or translating it along) the first joint axis. However, if there are obstacles or joint limits, the generic planar section is not sufficient for a complete analysis of the workspace.

In general, the workspace is a 6-dimensional space, which is difficult to handle. However, we can study its projection in the 3-dimensional position space.

5.7.2. *Singularity branches*

The *singularity branches* are the connected components of the set of singular configurations of Q. Since the singularities are always independent of the first joint, we can represent them in the joint space excluding the first joint. They are represented by surfaces of Q. However, for some particular cases, they can be reduced to subspaces of fewer dimensions (curves or points for example), which do not have a boundary in Q.

For the two degree-of-freedom planar robot with revolute joints shown in Figure 5.5, the determinant of the Jacobian matrix is equal to L1 L2 S2. The singularity branches, assuming unlimited joint ranges, are defined by the lines $\theta_2 = 0$ and $\theta_2 = \pm \pi$ (Figure 5.6). The corresponding workspace is presented in Figure 5.7.

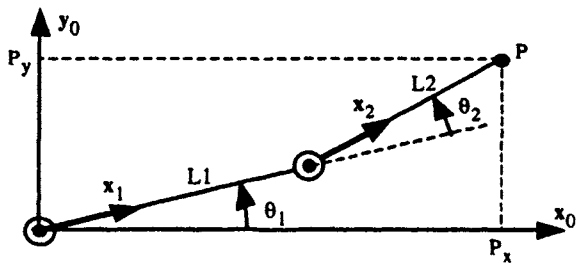

Figure 5.5. *Two degree-of-freedom planar robot*

For the Stäubli RX-90 robot, the joint space is partitioned by three singularity surfaces C3 = 0, S23 RL4 – C2 D3 = 0 and S5 = 0. Figure 5.8 shows these surfaces in the (θ_2, θ_3, θ_5) space and in the (θ_2, θ_3) plane.

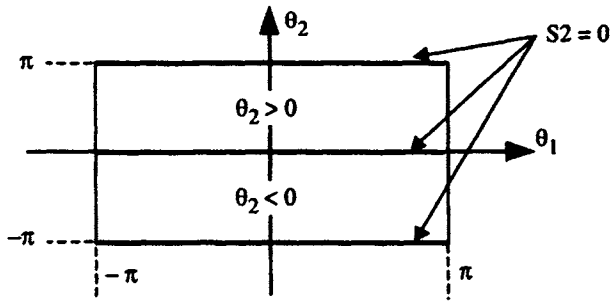

Figure 5.6. *Singularity branches of the planar robot with unlimited joints*

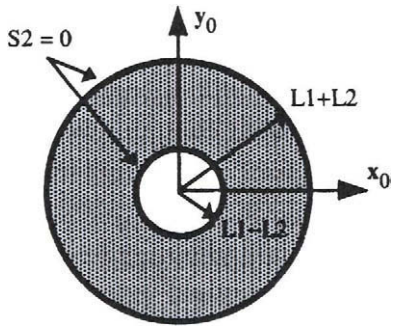

Figure 5.7. *Workspace of the planar robot with unlimited joints (L1>L2)*

5.7.3. *Jacobian surfaces*

Mapping the singularities into the workspace generally leads to surfaces (or subspaces with fewer dimensions) called *Jacobian surfaces*. These surfaces divide W into regions where the number of solutions of the IGM is constant and even [Roth 76], [Kholi 85], [Burdick 88]. In the presence of joint limits, additional boundaries appear in W, which define new regions in which the number of solutions of the IGM is always constant but not necessarily even [Spanos 85]. The Jacobian surfaces can be defined as the set of points in W where the IGM has at least two identical solutions [Kholi 87], [Spanos 85]. When the robot has three identical solutions for a point of the Jacobian surface, the robot is said to be *cuspidal* [El Omri 96].

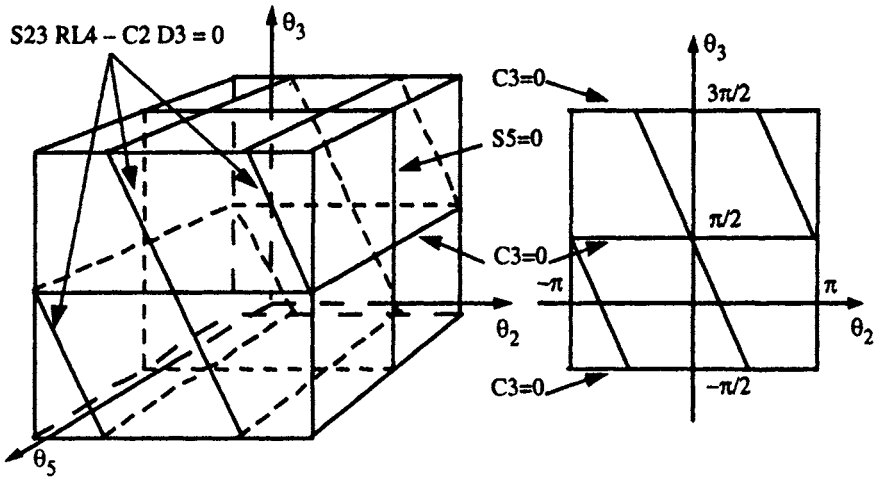

Figure 5.8. *Singularity branches of the Stäubli RX-90 robot*

In the case of a three degree-of-freedom robot, if the Jacobian surfaces are subspaces of fewer dimensions (for example a curve or a set of isolated points), the IGM for these points has an infinite number of solutions.

For the two degree-of-freedom planar robot shown in Figure 5.5, the Jacobian surfaces correspond to the singular configurations "extended arm" and "folded arm". They are represented by the circles with radii L1 + L2 and L1 – L2 respectively (Figure 5.7).

For the anthropomorphic shoulder of the Stäubli RX-90 robot, the Jacobian surfaces in the position workspace are of two types (Figure 5.9). The first is associated with the singular configurations where the point O_6 lies on the axis of the first joint. Their reciprocal mapping in **Q** give the singularity surfaces defined by S23RL4 – C2D3 = 0. For any point of these configurations, the IGM has an infinite number of solutions since θ_1 can be chosen arbitrarily. The other type of Jacobian surface corresponds to the singular configuration C3 = 0, and is represented by the surfaces of the spheres whose center is O_0, with radii D3 + RL4 ("extended arm" configuration) and D3 – RL4 ("folded arm" configuration) defining the external and internal boundaries of the workspace respectively. For the Stäubli RX-90 robot, the internal sphere is reduced to a point because D3 = RL4.

5.7.4. *Concept of aspect*

The concept of aspect has been introduced by Borrel [Borrel 86]. The aspects are the connected regions of the joint space inside which no minor of order M extracted from the Jacobian matrix **J** is zero, except if this minor is zero everywhere in the

joint domain. For a non-redundant robot manipulator, the only minor of order M is the Jacobian matrix itself. Therefore, the aspects are limited by the singularity branches and the joint limits (Figures 5.6 and 5.8). Consequently, they represent the maximum singularity-free regions.

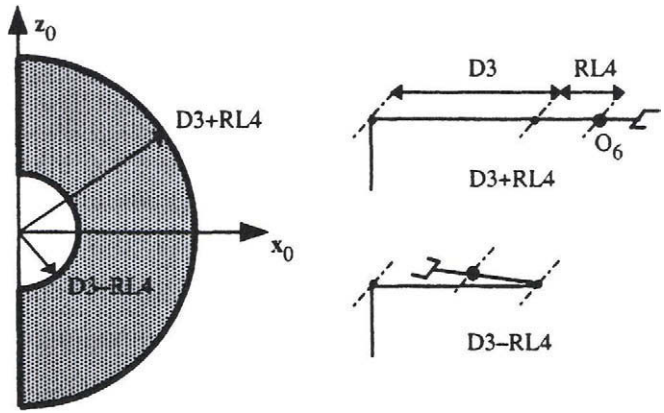

Figure 5.9. *Generic section of the workspace of an anthropomorphic shoulder with unlimited joints*

For a long time, it has been thought that the aspects also represent the uniqueness domains of the IGM solutions. Although this is indeed the case for most industrial robots with simple architectures, which are classified as non-cuspidal robots [El Omri 96], the IGM of cuspidal robots can have several solutions in the same aspect. Thus, a cuspidal robot can move from one IGM solution to another without encountering a singularity. Figure 5.10 shows a cuspidal robot with three revolute joints whose successive axes are perpendicular. The inverse geometric solution of the point X (Figure 5.11a) whose coordinates are $P_x = 2.5$, $P_y = 0$, $Pz = 0$ is given by the following four configurations (in degrees):

$$\mathbf{q}^{(1)} = [\,-101.52 \;\; -158.19 \;\; 104.88\,]^T,\ \mathbf{q}^{(2)} = [\,-50.92 \;\; -46.17 \;\; 141.16\,]^T$$
$$\mathbf{q}^{(3)} = [\,-164.56 \;\; -170.02 \;\; -12.89\,]^T,\ \mathbf{q}^{(4)} = [\,10.13 \;\; -22.33 \;\; -106.28\,]^T$$

The joint space of this robot is divided into two aspects (Figure 5.11a). We notice that the configurations $\mathbf{q}^{(2)}$ and $\mathbf{q}^{(3)}$ are located in the same aspect whereas $\mathbf{q}^{(1)}$ and $\mathbf{q}^{(4)}$ fall in the other aspect.

For cuspidal robots, the uniqueness domains of the IGM in the joint space are separated by the *characteristic surfaces* [Wenger 92], which are defined as the mapping of the Jacobian surfaces in the joint space using the IGM. Figure 5.11b

shows the singularities and the characteristic surfaces of the shoulder structure of Figure 5.10.

There is no general simple rule to identify the architectures of non-cuspidal robots. However, Table 5.1 gives a list of non-cuspidal shoulders as presented in [Wenger 93], [Wenger 98].

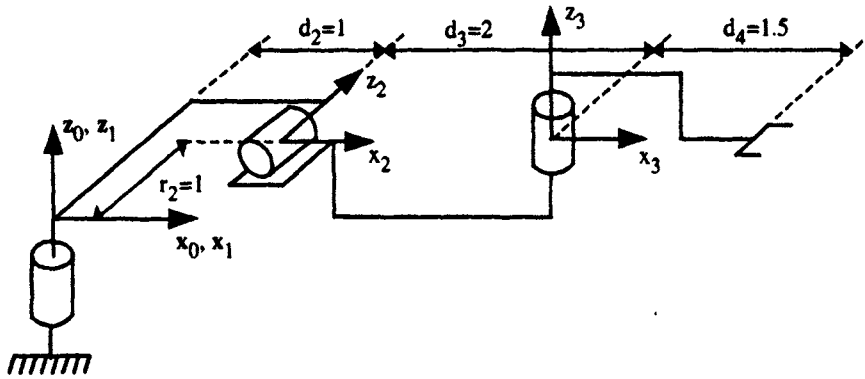

Figure 5.10. *Example of a cuspidal shoulder [Wenger 92]*

Table 5.1. *Non-cuspidal shoulders*

PPP	RPP	PRP	PPR	RRR	PRR	RPR	RRP
all	all	all	all	$s\alpha_2=0$ $s\alpha_3=0$ $d_2=0$ $d_3=0$ ($c\alpha_2=0$, $r_2=0$ and $r_3=0$)	$c\alpha_2=0$ $s\alpha_3=0$ $d_3=0$ ($s\alpha_2=0$ and $r_3=0$) ($s\alpha_2=0$ and $c\alpha_3=0$)	$c\alpha_2=0$ $c\alpha_3=0$ $s2=0$ $d_3+d_2c2=0$	$s\alpha_2=0$ $c\alpha_3=0$ $d_2=0$ ($c\alpha_2=0$, $s\alpha_3=0$ and $r_2=0$) $d_3+d_4c3=0$

5.7.5. *t-connected subspaces*

The t-connected subspaces are the regions in the workspace where any continuous trajectory can be followed by the robot end-effector. These subspaces are the mapping of the uniqueness domains in W using the DGM. For the non-cuspidal robots, the largest t-connected subspaces are the mapping of the aspects (and more generally of the free connected regions of the aspects when the environment is cluttered with obstacles [Wenger 89]). We do not present here the definition of the t-connected subspaces for the cuspidal robots. The interested reader can refer to [El Omri 96].

For the two degree-of-freedom planar robot shown in Figure 5.5, the straight line S2 = 0 separates the joint space domain into two aspects (Figure 5.12a) corresponding to the two solutions of the IGM, $\theta_2 > 0$ and $\theta_2 < 0$.

The mapping of these aspects in the workspace is identical if the joint ranges are equal to 2π. Figure 5.12b shows, for certain joint limits θ_{imax} and θ_{imin}, the t-connected regions: the hatched and non-hatched zones represent the mapping of the aspects $\theta_2 > 0$ and $\theta_2 < 0$ respectively. The trajectory PP' is located in the region mapped by the aspect $\theta_2 < 0$: thus it can only be realized if the initial configuration of the robot is $\theta_2 < 0$. Otherwise, one of the joints reaches its limit before arriving at the final position.

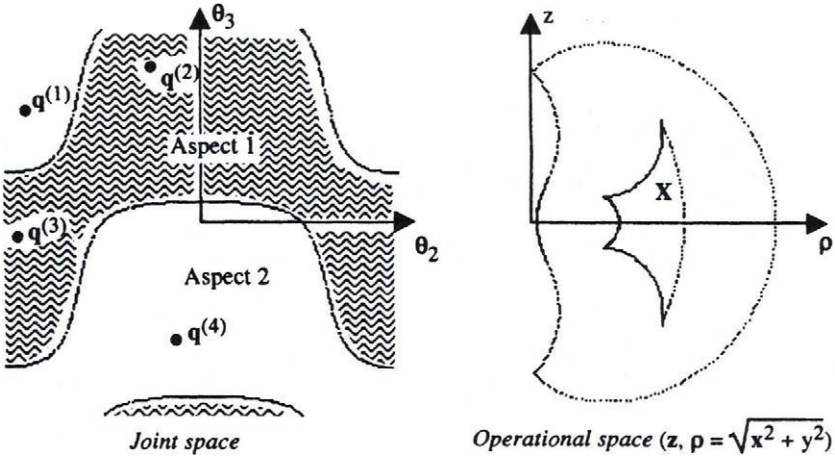

Figure 5.11a. *Aspects and workspace of the cuspidal shoulder of Figure 5.10*

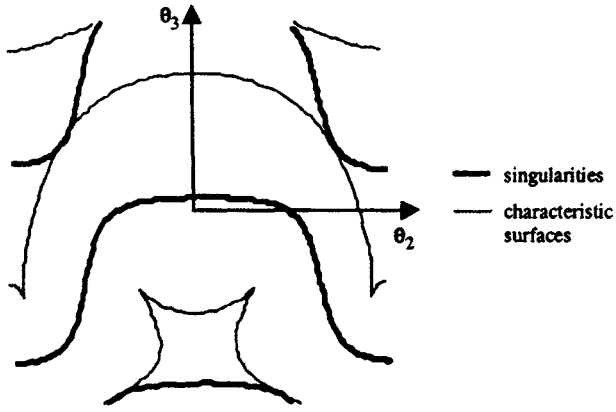

Figure 5.11b. *Singularity branches and characteristic surfaces of the cuspidal shoulder*

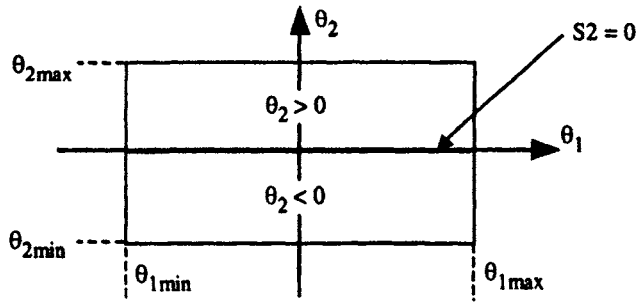

Figure 5.12a. *Aspects in the presence of joint limits*

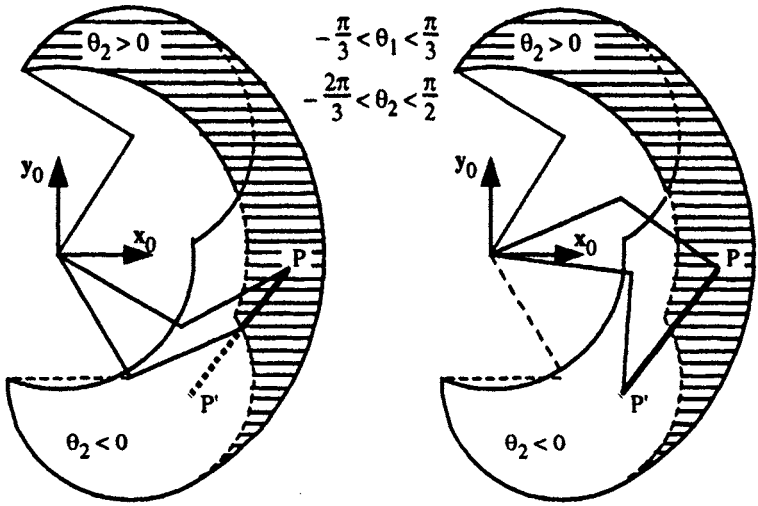

Figure 5.12b. *t-connected regions in the workspace*

5.8. Velocity transmission between joint space and task space

5.8.1. *Singular value decomposition*

At a given configuration, the (mxn) matrix **J** represents a linear mapping of the joint space velocities into the task space velocities. For simplicity, we write the basic Jacobian matrix $\mathbf{J}_n$ as **J**. When the end-effector coordinates are independent, we have n = N and m = M.

The singular value decomposition (SVD) theory states that for any (mxn) matrix **J** of rank r [Lawson 74], [Dongarra 79], [Klema 80], there exist orthogonal matrices **U** and **V** of dimensions (mxm) and (nxn) respectively such that:

$$\mathbf{J} = \mathbf{U}\,\boldsymbol{\Sigma}\,\mathbf{V}^T \qquad [5.27]$$

The (mxn) matrix $\boldsymbol{\Sigma}$ has the following form:

$$\boldsymbol{\Sigma} = \begin{bmatrix} \mathbf{S}_{rxr} & \mathbf{0}_{rx(n-r)} \\ \mathbf{0}_{(m-r)xr} & \mathbf{0}_{(m-r)x(n-r)} \end{bmatrix} \qquad [5.28]$$

S is an (rxr) diagonal matrix, formed by the non-zero singular values of **J**, which are arranged in decreasing order such that $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r$. The singular values of **J** are the square roots of the eigenvalues of the matrix $\mathbf{J}^T\mathbf{J}$ if $n \geq m$ (or $\mathbf{J}\,\mathbf{J}^T$ if $n \leq m$). The columns of **V** are the eigenvectors of and are called *right singular vectors* or *input vectors* of **J**. The columns of **U** are the eigenvectors of $\mathbf{J}\,\mathbf{J}^T$ and are called *left singular vectors* or *output vectors.*

Using equation [5.27], the kinematic model becomes:

$$\dot{\mathbf{X}} = \mathbf{U}\,\boldsymbol{\Sigma}\,\mathbf{V}^T\,\dot{\mathbf{q}} \qquad [5.29]$$

Since $\sigma_i = 0$ for $i > r$, we can write:

$$\dot{\mathbf{X}} = \sum_{i=1}^{r} \sigma_i\,\mathbf{U}_i\,\mathbf{V}_i^T\,\dot{\mathbf{q}} \qquad [5.30]$$

From equation [5.30], we deduce that (Figure 5.13):

- the vectors $\mathbf{V}_1, \ldots, \mathbf{V}_r$ form an orthonormal basis for the subspace of $\dot{\mathbf{q}}$ generating an end-effector velocity;
- the vectors $\mathbf{V}_{r+1}, \ldots, \mathbf{V}_n$ form an orthonormal basis for the subspace of $\dot{\mathbf{q}}$ giving $\dot{\mathbf{X}} = 0$. In other words, they define the null space of **J**, denoted by $\mathcal{N}(\mathbf{J})$;
- the vectors $\mathbf{U}_1, \ldots, \mathbf{U}_r$ form an orthonormal basis for the set of the achievable end-effector velocities $\dot{\mathbf{X}}$. Hence, they define the range space of **J**, denoted by $\mathcal{R}(\mathbf{J})$;
- the vectors $\mathbf{U}_{r+1}, \ldots, \mathbf{U}_m$ form an orthonormal basis for the subspace composed of the set of $\dot{\mathbf{X}}$ that cannot be generated by the robot. In other words, they define the complement of the range space, denoted by $\mathcal{R}(\mathbf{J})^{\perp}$;
- the singular values represent the velocity transmission ratio from the joint space to the task space. In fact, multiplying equation [5.30] by $\mathbf{U}_i^T$ yields:

$$\mathbf{U}_i^T\,\dot{\mathbf{X}} = \sigma_i\,\mathbf{V}_i^T\,\dot{\mathbf{q}} \quad \text{for } i = 1, \ldots, r \qquad [5.31]$$

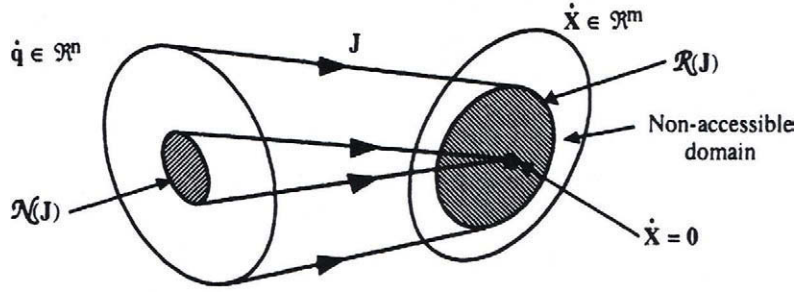

Figure 5.13. *Null space and range space of J (from [Asada 86])*

– since $\mathbf{J}^T = \mathbf{V}\,\mathbf{\Sigma}\,\mathbf{U}^T$, we deduce that:

$$\Re^m = \mathcal{R}(\mathbf{J}) + \mathcal{R}(\mathbf{J})^{\perp} = \mathcal{R}(\mathbf{J}) + \mathcal{N}(\mathbf{J}^T)$$
$$\Re^n = \mathcal{R}(\mathbf{J}^T) + \mathcal{N}(\mathbf{J})$$

5.8.2. *Velocity ellipsoid: velocity transmission performance*

The velocity transmission performance of a mechanism can be evaluated through the kinematic model [5.1]. Let us suppose that the joint velocities are limited such that:

$$-\dot{\mathbf{q}}_{max} \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}_{max} \qquad [5.32]$$

At a given configuration $\mathbf{q}$, the task space velocity satisfying these conditions belongs to:

$$\dot{\mathbf{X}}_{min} \leq \dot{\mathbf{X}} \leq \dot{\mathbf{X}}_{max} \qquad [5.33]$$

with:

$$\dot{\mathbf{X}}_{max} = \max(\mathbf{J}(\mathbf{q})\,\dot{\mathbf{q}}) \qquad [5.34]$$

$$\dot{\mathbf{X}}_{min} = \min(\mathbf{J}(\mathbf{q})\,\dot{\mathbf{q}}) \qquad [5.35]$$

Thus, the set of possible joint velocities (equation [5.32]) can be represented geometrically by a hyper-parallelepiped in the joint space. Equation [5.33] can also be represented by a hyper-parallelepiped in the task space. In this section, we develop another common approach to studying the velocity transmission between the joint space and the task space using an analytical ellipsoidal representation.

Let us consider the joint velocities contained in the unit sphere of the joint velocity space, such that [Yoshikawa 84b]:

$$\dot{\mathbf{q}}^T \dot{\mathbf{q}} \leq 1 \qquad [5.36]$$

We can show that the corresponding velocities in the task space are defined by the ellipsoid:

$$\dot{\mathbf{X}}^T (\mathbf{J}\,\mathbf{J}^T)^{-1} \dot{\mathbf{X}} \leq 1 \qquad [5.37]$$

The velocity ellipsoid is a useful tool for analyzing the velocity transmission performance of a robot at a given configuration. It is called the *manipulability ellipsoid*. The principal axes of the ellipsoid are given by the vectors $\mathbf{U}_1, \ldots, \mathbf{U}_m$, which are the eigenvectors of $\mathbf{J}\,\mathbf{J}^T$. The lengths of the principal axes are determined by the singular values $\sigma_1, \ldots, \sigma_m$ of $\mathbf{J}$. The optimum direction to generate velocity is along the major axis where the transmission ratio is maximum. Conversely, the velocity is most accurately controlled along the minor axis. Figure 5.14 shows the velocity ellipsoid for a 2R planar mechanism.

The volume of the velocity ellipsoid of a robot gives a measurement of its capacity to generate velocity. Consequently, we define the velocity manipulability of a robot as:

$$w(\mathbf{q}) = \sqrt{\det[\mathbf{J}(\mathbf{q})\,\mathbf{J}^T(\mathbf{q})]} \qquad [5.38]$$

For a non-redundant robot, this expression becomes:

$$w(\mathbf{q}) = |\det[\mathbf{J}(\mathbf{q})]| \qquad [5.39]$$

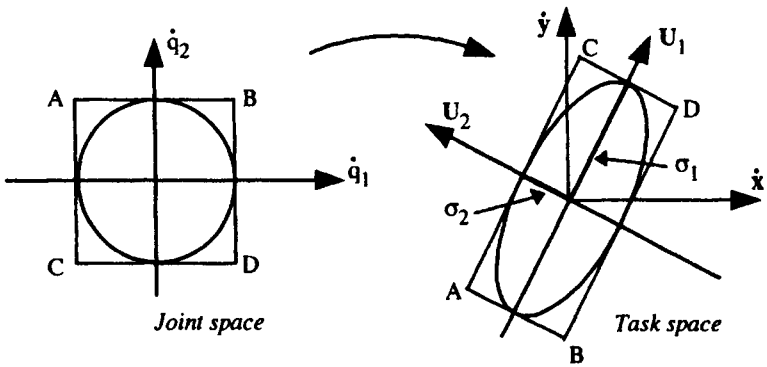

Figure 5.14. *Velocity ellipsoid for a two degree-of-freedom planar robot*

5.9. Static model

In this section, we establish the static model, which provides the joint torques (for revolute joints) or forces (for prismatic joints) corresponding to the wrench (forces and moments) exerted by the end-effector on the environment. We also discuss the duality between the kinematic model and the static model.

5.9.1. *Representation of a wrench*

Let us recall (§ 2.6) that a wrench $\mathbb{f}_i$ is represented by the screw, which is composed of a force $\mathbf{f}_i$ and a moment $\mathbf{m}_i$:

$$\mathbb{f}_i = \begin{bmatrix} \mathbf{f}_i \\ \mathbf{m}_i \end{bmatrix} \qquad [5.40]$$

We assume, unless otherwise stated, that the moment is defined about the point O_i, origin of frame R_i. Let the static wrench $\mathbb{f}_{en}$ to be exerted on the environment be defined as:

$$\mathbb{f}_{en} = \begin{bmatrix} \mathbf{f}_{en} \\ \mathbf{m}_{en} \end{bmatrix} = [\ f_x \quad f_y \quad f_z \quad m_x \quad m_y \quad m_z\ ]^T \qquad [5.41]$$

The subscript n indicates that the wrench is expressed at the origin O_n of frame R_n.

5.9.2. *Mapping of an external wrench into joint torques*

To compute the joint torques and forces Γ_e of a serial robot such that its end-effector can exert a static wrench $\mathbb{f}_{en}$, we make use of the principle of virtual work, which states that:

$$\Gamma_{en}^T \mathbf{dq}^* = \mathbb{f}_{en}^T \begin{bmatrix} \mathbf{dP}_n^* \\ \boldsymbol{\delta}_n^* \end{bmatrix} \qquad [5.42]$$

where the superscript (*) indicates virtual displacements.

Substituting $\mathbf{dP}_n^*$ and $\boldsymbol{\delta}_n^*$ from equation [5.4b] gives:

$$\Gamma_e = J_n^T \mathbb{f}_{en} \qquad [5.43]$$

We can use either the Jacobian matrix ${}^nJ_n$ or ${}^0J_n$ depending on whether the wrench $\mathbb{f}_{en}$ is referred to frame R_n or frame R_0 respectively.

5.9.3. *Velocity-force duality*

The Jacobian matrix appearing in the static model (equation [5.43]) is the same as that used in the differential or kinematic model. By analogy with the velocity transmission analysis (§ 5.8.1), we deduce the following results (Figure 5.15) [Asada 86]:

- the torques of the actuators are uniquely determined for an arbitrary wrench $\mathbb{f}$; the range space of J^T, denoted as $\mathcal{R}(J^T)$, is the set of Γ balancing the static wrench $\mathbb{f}$ according to equation [5.43];
- for a zero Γ, the corresponding static wrench can be non-zero; we thus define the null space of J^T, $\mathcal{N}(J^T)$, as the set of static wrenches that do not require actuator torques in order to be balanced. In this case, the endpoint wrench is borne by the structure of the robot. Note that the null space of J^T, $\mathcal{N}(J^T)$, which is the orthogonal complement of $\mathcal{R}(J)$, also represents the set of directions along which the robot cannot generate velocity;
- some joint torques Γ cannot be compensated by $\mathbb{f}$. These torques correspond to the vectors of the null space $\mathcal{N}(J)$, orthogonal complement of the space $\mathcal{R}(J^T)$.

The basis of these spaces can be defined using the columns of the matrices U and V of the singular value decomposition of J as indicated for the velocity case (§ 5.8.1).

Analogously, we can study the force transmission performance using a force manipulability ellipsoid, which corresponds to the set of achievable wrench in the task space $\mathfrak{R}^m$ corresponding to the constraint $\Gamma^T \Gamma \leq 1$. Thus, the force ellipsoid is defined by $\mathbb{f}^T J J^T \mathbb{f} \leq 1$. Consequently, we can deduce that the velocity ellipsoid (equation [5.37]) and the force ellipsoid have the same principal axes but the axis lengths are reciprocal (Figure 5.16). This means that the optimum direction for generating velocity is the optimum direction for controlling force. Similarly, the optimal direction for exerting force is also the optimum direction for controlling velocity.

From the control point of view, this behavior makes sense: the velocity is controlled most accurately in the direction where the robot can resist large force

disturbances, and force is most accurately controlled in the direction where the robot can rapidly adapt its motion.

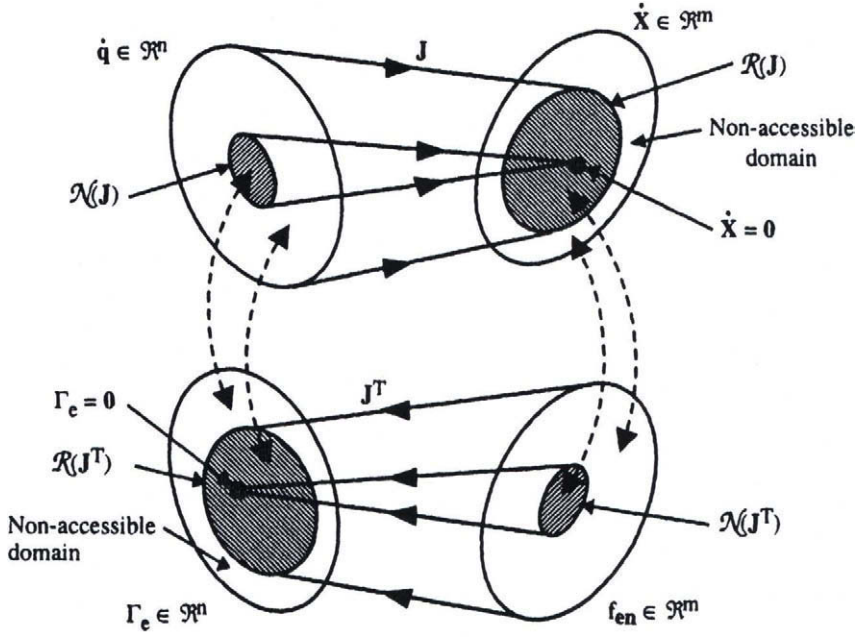

Figure 5.15. *Velocity-force duality (from [Asada 86])*

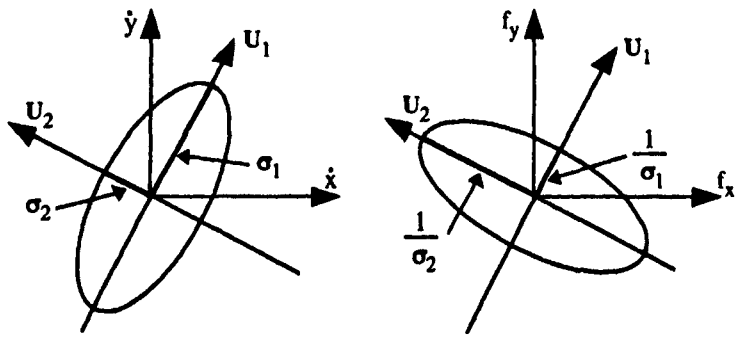

Figure 5.16. *Velocity and force ellipsoids*

5.10. Second order kinematic model

The second order kinematic model allows us to compute the acceleration of the end-effector in terms of positions, velocities and accelerations of the joints. By differentiating equation [5.1] with respect to time, we obtain the following expression:

$$\ddot{\mathbf{X}} = \mathbf{J}\,\ddot{\mathbf{q}} + \dot{\mathbf{J}}\,\dot{\mathbf{q}} \qquad [5.44]$$

where:

$$\dot{\mathbf{J}}(\mathbf{q}, \dot{\mathbf{q}}) = \frac{d}{dt}\mathbf{J}(\mathbf{q}) \qquad [5.45]$$

Using the basic Jacobian matrix, the second order kinematic model can be written as:

$$\begin{bmatrix} \dot{\mathbf{V}}_n \\ \dot{\boldsymbol{\omega}}_n \end{bmatrix} = \mathbf{J}_n\,\ddot{\mathbf{q}} + \dot{\mathbf{J}}_n\,\dot{\mathbf{q}} \qquad [5.46]$$

However, it is most efficient from the computational cost point of view to obtain $\dot{\mathbf{V}}_n$ and $\dot{\boldsymbol{\omega}}_n$ from the following recursive equations, for j = 1, ..., n, which will be developed in Chapter 9:

$$\begin{cases} {}^j\dot{\boldsymbol{\omega}}_j = {}^jA_{j-1}\,{}^{j-1}\dot{\boldsymbol{\omega}}_{j-1} + \bar{\sigma}_j\,(\ddot{q}_j\,{}^j\mathbf{a}_j + {}^j\boldsymbol{\omega}_{j-1} \times \dot{q}_j\,{}^j\mathbf{a}_j) \\ {}^jU_j = {}^j\dot{\hat{\boldsymbol{\omega}}}_j + {}^j\hat{\boldsymbol{\omega}}_j\,{}^j\hat{\boldsymbol{\omega}}_j \\ {}^j\dot{\mathbf{V}}_j = {}^jA_{j-1}\,({}^{j-1}\dot{\mathbf{V}}_{j-1} + {}^{j-1}U_{j-1}\,{}^{j-1}\mathbf{P}_j) + \sigma_j\,(\ddot{q}_j\,{}^j\mathbf{a}_j + 2{}^j\boldsymbol{\omega}_{j-1} \times \dot{q}_j\,{}^j\mathbf{a}_j) \end{cases} \qquad [5.47]$$

The angular velocities ${}^j\boldsymbol{\omega}_{j-1}$ and ${}^j\boldsymbol{\omega}_j$ are calculated using equation [5.23].

In certain applications, such as the control in the task space (§ 14.4.3), we need to compute the vector $\dot{\mathbf{J}}\,\dot{\mathbf{q}}$. Instead of taking the derivative of $\mathbf{J}$ with respect to time and multiplying by $\dot{\mathbf{q}}$, it is more efficient to make use of the recursive equations [5.47] with $\ddot{\mathbf{q}}$ equal to zero in order to leave out the terms involving $\ddot{\mathbf{q}}$ [Khalil 87a].

5.11. Kinematic model associated with the task coordinate representation

Let $\mathbf{X} = \begin{bmatrix} \mathbf{X}_p \\ \mathbf{X}_r \end{bmatrix}$ be any representation of the location of frame R_n relative to frame R_0, where $\mathbf{X}_p$ and $\mathbf{X}_r$ denote the position and orientation vectors respectively. The relationships between the velocities $\dot{\mathbf{X}}_p$ and $\dot{\mathbf{X}}_r$ and the velocities ${}^0\mathbf{V}_n$ and ${}^0\boldsymbol{\omega}_n$ of frame R_n are given as:

$$\begin{cases} \dot{\mathbf{X}}_p = \boldsymbol{\Omega}_p \, {}^0\mathbf{V}_n \\ \dot{\mathbf{X}}_r = \boldsymbol{\Omega}_r \, {}^0\boldsymbol{\omega}_n \end{cases} \qquad [5.48]$$

Similar relations can be derived to express the differential vectors $\mathbf{dX}_p$ and $\mathbf{dX}_r$ as functions of the vectors ${}^0\mathbf{dP}_n$ and ${}^0\boldsymbol{\delta}_n$:

$$\begin{cases} \mathbf{dX}_p = \boldsymbol{\Omega}_p \, {}^0\mathbf{dP}_n \\ \mathbf{dX}_r = \boldsymbol{\Omega}_r \, {}^0\boldsymbol{\delta}_n \end{cases} \qquad [5.49]$$

In matrix form, equation [5.48] becomes:

$$\begin{bmatrix} \dot{\mathbf{X}}_p \\ \dot{\mathbf{X}}_r \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Omega}_p & \mathbf{0}_3 \\ \mathbf{0}_3 & \boldsymbol{\Omega}_r \end{bmatrix} \begin{bmatrix} {}^0\mathbf{V}_n \\ {}^0\boldsymbol{\omega}_n \end{bmatrix} = \boldsymbol{\Omega} \begin{bmatrix} {}^0\mathbf{V}_n \\ {}^0\boldsymbol{\omega}_n \end{bmatrix} \qquad [5.50]$$

Using equation [5.4a], we deduce that:

$$\begin{bmatrix} \dot{\mathbf{X}}_p \\ \dot{\mathbf{X}}_r \end{bmatrix} = \boldsymbol{\Omega} \, {}^0\mathbf{J}_n \, \dot{\mathbf{q}} = \mathbf{J}_x \, \dot{\mathbf{q}} \qquad [5.51]$$

with:

$$\mathbf{J}_x = \boldsymbol{\Omega} \, {}^0\mathbf{J}_n \qquad [5.52]$$

The matrix $\boldsymbol{\Omega}_p$ is equal to $\mathbf{I}_3$ when the position of frame R_n is described by the Cartesian coordinates.

In this section, we show how to calculate $\boldsymbol{\Omega}_r$ and $\boldsymbol{\Omega}_r^{-1}$ for different orientation representations. These expressions are necessary for establishing the kinematic

model corresponding to the representation at hand. When the orientation description is not redundant, the inverse of Ω can be written as:

$$\Omega^{-1} = \begin{bmatrix} I_3 & 0_3 \\ 0_3 & \Omega_r^{-1} \end{bmatrix} \qquad [5.53]$$

If the description of the orientation is redundant, which is the case with the direction cosines and the quaternions (Euler parameters), the matrices Ω_r, and consequently Ω, are rectangular. We then use the so-called left inverse, which is a particular case of the pseudoinverse (Appendix 4). The left inverse is defined by:

$$\Omega^{+} = \begin{bmatrix} I_3 & 0_3 \\ 0_3 & \Omega_r^{+} \end{bmatrix} \qquad [5.54]$$

with:

$$\begin{cases} \Omega^{+} = (\Omega^T \Omega)^{-1} \Omega^T \\ \Omega^{+} \Omega = I_6 \end{cases} \qquad [5.55]$$

Such a matrix exists if Ω is of rank 6, which means that Ω_r is of rank 3.

5.11.1. *Direction cosines*

The velocity of the vectors **s**, **n**, **a** are given by:

$$\begin{cases} {}^0\dot{\mathbf{s}}_n = {}^0\omega_n \times {}^0\mathbf{s}_n \\ {}^0\dot{\mathbf{n}}_n = {}^0\omega_n \times {}^0\mathbf{n}_n \\ {}^0\dot{\mathbf{a}}_n = {}^0\omega_n \times {}^0\mathbf{a}_n \end{cases} \qquad [5.56]$$

Using the vector product operator defined in [2.32], equations [5.56] can be written in the following matrix form [Khatib 80]:

$$\dot{X}_r = \begin{bmatrix} {}^0\dot{\mathbf{s}}_n \\ {}^0\dot{\mathbf{n}}_n \\ {}^0\dot{\mathbf{a}}_n \end{bmatrix} = \begin{bmatrix} -{}^0\hat{\mathbf{s}}_n \\ -{}^0\hat{\mathbf{n}}_n \\ -{}^0\hat{\mathbf{a}}_n \end{bmatrix} {}^0\omega_n = \Omega_{CD}\, {}^0\omega_n \qquad [5.57]$$

where Ω_{CD} is a (9x3) matrix. To calculate Ω_{CD}^{+}, we use the fact that:

$$\Omega_{CD}^{T}\,\Omega_{CD} = 2\,I_3 \qquad [5.58]$$

Using equation [5.55] and taking into account that the matrices $\hat{\mathbf{s}}$, $\hat{\mathbf{n}}$, $\hat{\mathbf{a}}$ are skew-symmetric, we obtain:

$$\Omega_{CD}^{+} = \frac{1}{2}\Omega_{CD}^{T} = \frac{1}{2}\left[\ {}^{0}\hat{\mathbf{s}}_n \quad {}^{0}\hat{\mathbf{n}}_n \quad {}^{0}\hat{\mathbf{a}}_n\ \right] \qquad [5.59]$$

5.11.2. *Euler angles*

We deduce from § 3.6.1 that ϕ is the rotation angle about $\mathbf{z}_0 = [\ 0\ \ 0\ \ 1\ ]^T$, θ is the rotation angle about the current **x** axis (after applying **rot**(**z**, ϕ)) whose unit vector with respect to R_0 is $[\ C\phi\ \ S\phi\ \ 0\ ]^T$, and ψ is the rotation angle about the current **z** axis (after applying **rot**(**z**, ϕ) **rot**(**x**, θ)) whose unit vector components with respect to R_0 are $[\ S\phi S\theta\ \ -C\phi S\theta\ \ C\theta\ ]^T$. Thus, the velocity of frame R_n relative to frame R_0 is given by:

$${}^{0}\omega_n = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\dot{\phi} + \begin{bmatrix} C\phi \\ S\phi \\ 0 \end{bmatrix}\dot{\theta} + \begin{bmatrix} S\phi S\theta \\ -C\phi S\theta \\ C\theta \end{bmatrix}\dot{\psi} \qquad [5.60]$$

thus:

$${}^{0}\omega_n = \begin{bmatrix} 0 & C\phi & S\phi S\theta \\ 0 & S\phi & -C\phi S\theta \\ 1 & 0 & C\theta \end{bmatrix}\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \qquad [5.61]$$

which we identify with:

$${}^{0}\omega_n = \Omega_{Eul}^{-1}\,\dot{\mathbf{X}}_r = \Omega_{Eul}^{-1}\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \qquad [5.62]$$

By taking the inverse of Ω_{Eul}^{-1}, we obtain:

$$\Omega_{Eul} = \begin{bmatrix} -S\phi\ \mathrm{cotg}\theta & C\phi\ \mathrm{cotg}\theta & 1 \\ C\phi & S\phi & 0 \\ S\phi/S\theta & -C\phi/S\theta & 0 \end{bmatrix} \qquad [5.63]$$

Ω_{Eul} is singular when $S\theta = 0$, as already obtained in § 3.6.1.

5.11.3. *Roll-Pitch-Yaw angles*

Similarly, we can write:

$$^0\omega_n = \begin{bmatrix} 0 & -S\phi & C\phi C\theta \\ 0 & C\phi & S\phi C\theta \\ 1 & 0 & -S\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \Omega_{RTL}^{-1} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \qquad [5.64]$$

from which we obtain:

$$\Omega_{RTL} = \begin{bmatrix} C\phi\ tg\theta & S\phi\ tg\theta & 1 \\ -S\phi & C\phi & 0 \\ C\phi/C\theta & S\phi/C\theta & 0 \end{bmatrix} \qquad [5.65]$$

This matrix is singular when $C\theta = 0$, as already obtained in § 3.6.2.

5.11.4. *Quaternions*

Differentiating equation [3.34] with respect to time and equating the diagonal elements with those of equation [5.56] leads to the following equation:

$$\begin{cases} 2(Q_1\dot{Q}_1 + Q_2\dot{Q}_2) = (Q_2Q_4 - Q_1Q_3)\omega_y - (Q_2Q_3 + Q_1Q_4)\omega_z \\ 2(Q_1\dot{Q}_1 + Q_3\dot{Q}_3) = (Q_2Q_3 - Q_1Q_4)\omega_z - (Q_3Q_4 + Q_1Q_2)\omega_x \\ 2(Q_1\dot{Q}_1 + Q_4\dot{Q}_4) = (Q_3Q_4 - Q_1Q_2)\omega_x - (Q_2Q_4 + Q_1Q_3)\omega_y \end{cases} \qquad [5.66]$$

By differentiating equation [3.31] with respect to time, we obtain:

$$Q_1\dot{Q}_1 + Q_2\dot{Q}_2 + Q_3\dot{Q}_3 + Q_4\dot{Q}_4 = 0 \qquad [5.67]$$

From equations [5.66] and [5.67], we deduce that:

$$\dot{\mathbf{X}}_r = \dot{\mathbf{Q}} = [\dot{Q}_1 \;\; \dot{Q}_2 \;\; \dot{Q}_3 \;\; \dot{Q}_4]^T = \boldsymbol{\Omega}_Q \, {}^0\boldsymbol{\omega}_n \qquad [5.68]$$

with:

$$\boldsymbol{\Omega}_Q = \frac{1}{2}\begin{bmatrix} -Q_2 & -Q_3 & -Q_4 \\ Q_1 & Q_4 & -Q_3 \\ -Q_4 & Q_1 & Q_2 \\ Q_3 & -Q_2 & Q_1 \end{bmatrix} \qquad [5.69]$$

To obtain the inverse relationship, we use the left inverse. While taking into account that $\boldsymbol{\Omega}_Q^T \boldsymbol{\Omega}_Q = \frac{1}{4}$, we obtain:

$$\boldsymbol{\Omega}_Q^+ = 4\,\boldsymbol{\Omega}_Q^T \qquad [5.70]$$

We note that, since the integration of the angular velocity ${}^0\boldsymbol{\omega}_n$ does not yield an orientation representation, equation [5.69] can be used to obtain $\dot{\mathbf{Q}}$ whose integration gives the orientation by the Quaternion representation.

5.12. Conclusion

In this chapter, we have shown how to obtain the kinematic model of a robot manipulator using the basic Jacobian matrix. This model allows us to compute the linear and angular velocities of the end-effector in terms of the joint velocities. The Jacobian matrix can be decomposed into two or three matrices containing simpler terms.

Then, we have shown how to use the Jacobian matrix to analyze the workspace and the velocity space of a robot. We have also demonstrated how to use the Jacobian matrix to obtain the static model and we have highlighted the duality of this model with the kinematic model. Finally, the kinematic models associated with the various representations of the task coordinates have been established.

The kinematic model can also be used to find a numerical solution to the inverse geometric problem for a general robot. The necessary tool to obtain this solution is the inverse kinematic model, which is the topic of the next chapter.