

机器人运动学讲义

机械臂基本结构

一个机械臂系统由一系列的连杆与关节构成：连杆连接两个关节，驱动器驱动关节转动，关节带动连杆位置发生变化。关节和连杆首尾相连，形成了联动的“链条”；链条的一端固定在底座，另一端可以自由运动。自由运动的这一端，通过装上末端执行器，以执行操作任务，如图 1 所示。

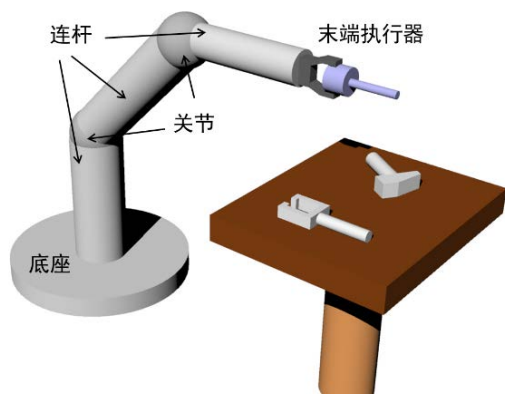


图 1. 机械臂固定在底座上，由一系列连杆与关节构成，末端安装任务执行机构

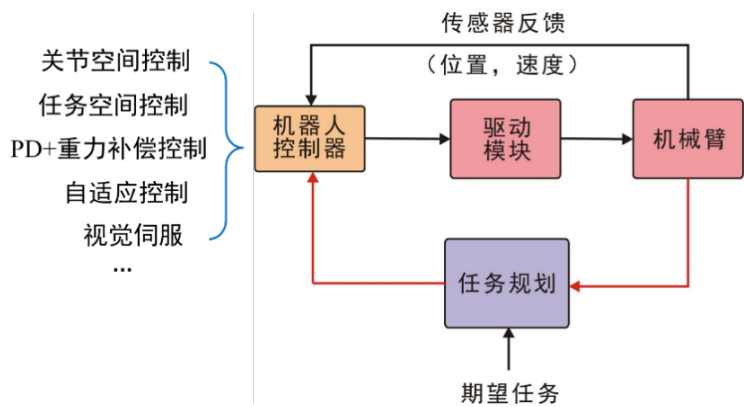


图 2. 机械臂结构框图

机械臂的典型结构如图 2 所示，包括：机器人控制器，驱动模块，机械臂本体（包含内部传感器），以及任务规划模块。机械臂的关节转角由内部传感器测量（如编码器）；任务规划模块根据期望任务生成机器人的参考轨迹；机器人控制器接收传感器反馈并与参考轨迹比较，生成控制输入，通过驱动模块驱动机械臂执行任务。在机器人控制器中，我们可以编程实现不同的控制算法，比如，关节空间控制、任务空间控制、自适应控制，视觉伺服等。

机械臂关节一般分为以下两种类型，如图 3 所示：

- 1) 旋转关节：沿着转轴进行旋转运动；
- 2) 平移关节：沿着轴线进行平移运动。

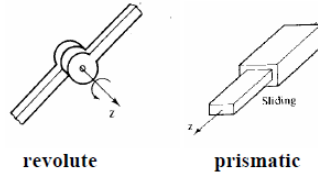


图 3. (a) 旋转关节 (b) 平移关节

对于一个机械臂系统，我们在不同的空间与坐标系来描述它的位置与姿态。比如，在关节空间引入关节转角向量：

$$q = [q_1, \dots, q_n]^T \in R^n \quad (1)$$

其中， q_1, q_2, q_3 分别是第一至第三个关节的转角， q 为包含了所有关节角度的向量；也可以通过引入末端执行器的位置和姿态进行刻画：

$$r = [r_1, \dots, r_n]^T \in R^6 \quad (2)$$

向量 r 前三维表示末端在笛卡尔空间的位置，后三维表示末端的旋转角度；如果采用相机直接测量末端执行器或者其他特征，也可以引入特征在 2D 图像空间的像素点位置进行刻画：

$$x = [u, v]^T \in R^2 \quad (3)$$

以上的笛卡尔空间和图像空间定义为任务空间，即可以在这些空间中直接定义机器人末端要执行的任务。机械臂执行操作任务是通过控制末端执行器的位置与角度来实现的。因此，我们首先定义机械臂的关节角度变化与末端执行器的关系，即图 4 表示的运动学模型。

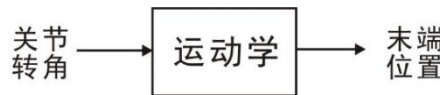


图 4. 运动学模型刻画了关节转角与末端执行器的关系

具体而言，机械臂的前向运动学问题定义为：给定机械臂各个关节转角，求解末端执行器相对于固定底座的位置与角度。

机械臂的运动学模型可以通过引入局部坐标系之间的转换矩阵进行描述，如图 5 所示。在图 5 中， T_0^1 表示从底座到第一个关节的转换矩阵， T_1^2 表示从第一到

第二个关节的转换矩阵，依次类推。在获取所有局部坐标系的转换矩阵之后，通过矩阵相乘，可以将末端的位置和角度表示在底座的坐标系内。

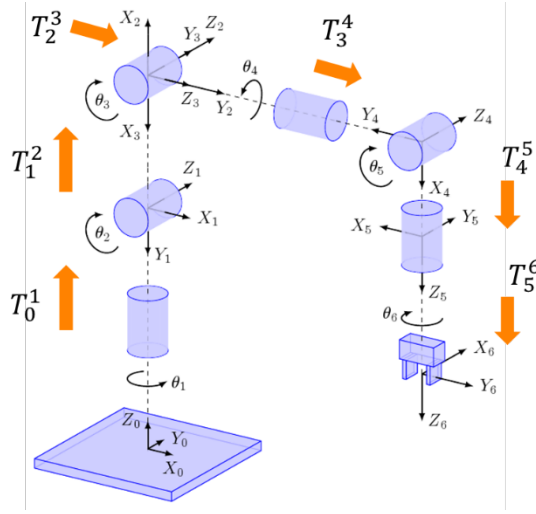


图 5. 各个关节对应的局部坐标系之间通过一系列的转换矩阵关联

一个简单的单关节前向运动学模型如图 6 所示：

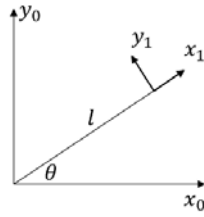


图 6. 单关节的前向运动学模型

其对应的前向运动学方程可描述为：

$$\begin{cases} x_0 = l \cdot \cos\theta \\ y_0 = l \cdot \sin\theta \end{cases} \quad (4)$$

其中， l 表示连杆长度， θ 表示关节转角。

坐标变换

前向运动学模型可通过坐标变换建立联系。假定三维空间中两个坐标系：F 坐标系（即固定坐标系 fixed）与 M 坐标系（即移动坐标系 mobile），如图 7 所示。假定该三维空间中存在一个点 p ，该点相对于 F 坐标系的位置为：

$$p = p_1^f f_1 + p_2^f f_2 + p_3^f f_3 \quad (5)$$

该点相对于 M 坐标系的位置可以描述为：

$$p = p_1^m m_1 + p_2^m m_2 + p_3^m m_3 \quad (6)$$

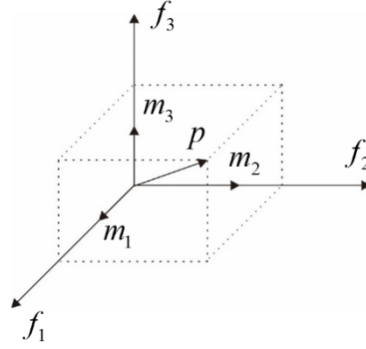


图 7. 向量 p 表示在三维空间中的 M 或 F 坐标系

在公式(5)中, f_1, f_2, f_3 表示坐标轴三个方向的单位向量, 表示为: $[1,0,0]$, $[0,1,0]$, $[0,0,1]$, p_1^f, p_2^f, p_3^f 表示 p 在这三个单位向量方向上的投影长度。比如, 如果将向量 $p=[2,3,2]$ 投影在 F 坐标系, 根据公式(5), 我们可以写成:

$$p = 2f_1 + 3f_2 + 2f_3 \quad (7)$$

同理, p_1^m, p_2^m, p_3^m 分别表示 p 在 M 坐标系三个单位向量方向上的投影长度。此外, 两个向量的投影关系也可以通过点乘来表示。比如, p 点乘 f_1 , 即为 p 在单位向量 f_1 的投影, 即 p_1^f 。因此, 我们可以得到:

$$\begin{cases} p_1^f = p \cdot f_1 \\ p_2^f = p \cdot f_2 \\ p_3^f = p \cdot f_3 \end{cases} \quad (8)$$

此外, 因此向量 p 也可以分别投影在 M 坐标系的三个单位向量上 (如公式(6))。将公式(6)代入方程组(8)的第一个公式可以得到:

$$p_1^f = p \cdot f_1 = p_1^m m_1 \cdot f_1 + p_2^m m_2 \cdot f_1 + p_3^m m_3 \cdot f_1 \quad (9)$$

同理, p_2^f, p_3^f 都可以写成这种形式。因此, 我们推导可得:

$$\begin{cases} p_1^f = p_1^m m_1 \cdot f_1 + p_2^m m_2 \cdot f_1 + p_3^m m_3 \cdot f_1 \\ p_2^f = p_1^m m_1 \cdot f_2 + p_2^m m_2 \cdot f_2 + p_3^m m_3 \cdot f_2 \\ p_3^f = p_1^m m_1 \cdot f_3 + p_2^m m_2 \cdot f_3 + p_3^m m_3 \cdot f_3 \end{cases} \quad (10)$$

把方程(10)两边整理下成矩阵与向量相乘的形式可得:

$$\begin{bmatrix} p_1^f \\ p_2^f \\ p_3^f \end{bmatrix} = \begin{bmatrix} m_1 \cdot f_1 & m_2 \cdot f_1 & m_3 \cdot f_1 \\ m_1 \cdot f_2 & m_2 \cdot f_2 & m_3 \cdot f_2 \\ m_1 \cdot f_3 & m_2 \cdot f_3 & m_3 \cdot f_3 \end{bmatrix} \begin{bmatrix} p_1^m \\ p_2^m \\ p_3^m \end{bmatrix} \quad (11)$$

其中，公式(11)等号右侧表示旋转矩阵：

$$R = \begin{bmatrix} m_1 \cdot f_1 & m_2 \cdot f_1 & m_3 \cdot f_1 \\ m_1 \cdot f_2 & m_2 \cdot f_2 & m_3 \cdot f_2 \\ m_1 \cdot f_3 & m_2 \cdot f_3 & m_3 \cdot f_3 \end{bmatrix} \quad (12)$$

因此，公式(11)可以表示成：

$$\begin{bmatrix} p_1^f \\ p_2^f \\ p_3^f \end{bmatrix} = R \begin{bmatrix} p_1^m \\ p_2^m \\ p_3^m \end{bmatrix} \quad (13)$$

其中，等式左边的向量表示 p 在 F 坐标系的投影，等式右边的向量表示 p 在 M 坐标系的投影。旋转矩阵 R 用来关联这些投影关系。

如果将坐标系 M 围绕坐标系 F 中某一轴进行旋转，旋转之后的 R 矩阵称为单位旋转矩阵。两个坐标系之间的旋转包括三种情况：绕 f_1 旋转 θ_1 ，绕 f_2 旋转 θ_2 ，绕 f_3 旋转 θ_3 ，如图 8 所示。

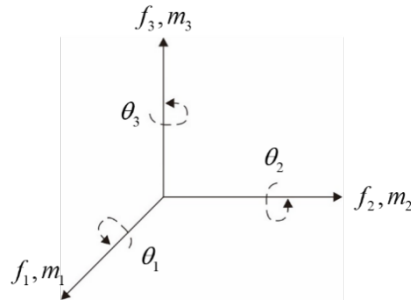


图 8. 两个坐标系之间的旋转

我们先考虑 M 坐标系绕 f_1 旋转 ϕ 角度，旋转完之后如图 9 所示：

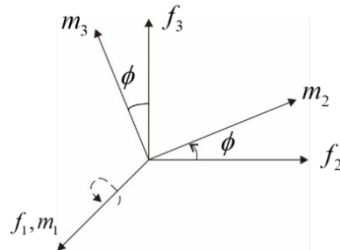


图 9. 绕 f_1 旋转角度 ϕ

旋转完之后： m_1 与 f_1 的点乘，因为是同一个向量，所以两者相乘为 1； m_2 ， m_3

与 f_1 垂直, f_2, f_3 与 m_1 垂直, 所以点乘为 0; m_2 与 f_2 的点乘为 $1 \cdot 1 \cdot \cos \phi$, ϕ 为 m_2 与 f_2 夹角; 矩阵 R 中的其他元素以此类推。最后得到的旋转矩阵如公式(14)所示。

$$\begin{bmatrix} p_1^f \\ p_2^f \\ p_3^f \end{bmatrix} = \underbrace{\begin{bmatrix} m_1 \cdot f_1 & m_2 \cdot f_1 & m_3 \cdot f_1 \\ m_1 \cdot f_2 & m_2 \cdot f_2 & m_3 \cdot f_2 \\ m_1 \cdot f_3 & m_2 \cdot f_3 & m_3 \cdot f_3 \end{bmatrix}}_{R_1(\phi)} \begin{bmatrix} p_1^m \\ p_2^m \\ p_3^m \end{bmatrix} \rightarrow R_1(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix} \quad (14)$$

将 M 坐标系统绕 f_2 旋转角度 ϕ , 可以得到:

$$R_2(\phi) = \begin{bmatrix} \cos \phi & 0 & \sin \phi \\ 0 & 1 & 0 \\ -\sin \phi & 0 & \cos \phi \end{bmatrix} \quad (15)$$

将 M 坐标系统绕 f_3 旋转角度 ϕ , 可以得到:

$$R_3(\phi) = \begin{bmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (16)$$

除了绕单一坐标轴旋转之外, 也可将 M 坐标系统顺序绕三轴进行复合旋转。比如, 先绕 f_1 旋转 θ_1 , 然后绕 f_2 旋转 θ_2 , 最后绕 f_3 旋转 θ_3 , 获得的最终旋转矩阵如公式(17)所示。其中, C_1 表示 $\cos(\theta_1)$, S_1 表示 $\sin(\theta_1)$, 依次类推。

$$YPR(\theta) = R_3(\theta_3)R_2(\theta_2)R_1(\theta_1) = \begin{bmatrix} C_2C_3 & S_1S_2C_3 - C_1S_3 & C_1S_2C_3 + S_1S_3 \\ C_2S_3 & S_1S_2S_3 + C_1C_3 & C_1S_2S_3 - S_1C_3 \\ -S_2 & S_1C_2 & C_1C_2 \end{bmatrix} \quad (17)$$

因为矩阵的乘法不满足交换律, 所以如果旋转的顺序发生变化, 最后得到的旋转矩阵也不一样。公式(17)中每一轴的旋转角度, 都有特定的名称: yaw, pitch, roll, 如图 10 所示。

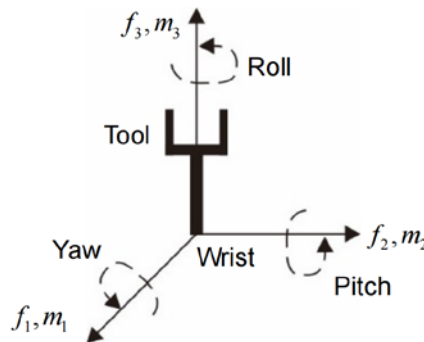


图 10. yaw-pitch-roll 复合旋转

除了坐标系之间的旋转, 两者还可能存在着平移的关系, 这两种基本的变换关系如图 11 所示。

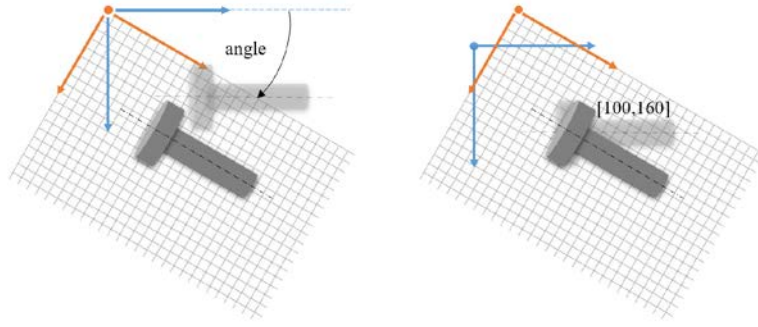


图 11. (a) 两坐标系成旋转关系 (b)两坐标系成旋转加平移的复合关系

旋转加平移的复合关系可以通过齐次矩阵 T 来描述：

$$T = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \quad (18)$$

在齐次矩阵 T 中，左上部分为旋转矩阵 R ，刻画两个坐标轴之间的旋转关系；右上为平移向量 p ，表示两个坐标系之间的平移关系；最右下角元素为 1，最后一行的其他元素为零。齐次矩阵 T 为方阵，根据公式(17)的定义，通过多个齐次矩阵的连乘可以刻画多个坐标系的连续变换关系。

D-H 理论

在图 5 所示的机械臂系统中，局部坐标系之间的转换关系是通过关节旋转来实现的。因此，机械臂系统中的齐次矩阵 T 可以通过 D-H 理论进一步细化为：

$$T_{k-1}^k = \begin{bmatrix} \cos(\theta_k) & -\cos(\alpha_k)\sin(\theta_k) & \sin(\alpha_k)\sin(\theta_k) & a_k\cos(\theta_k) \\ \sin(\theta_k) & \cos(\alpha_k)\cos(\theta_k) & -\sin(\alpha_k)\cos(\theta_k) & a_k\sin(\theta_k) \\ 0 & \sin(\alpha_k) & \cos(\alpha_k) & d_k \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (19)$$

其中， θ_k α_k a_k d_k 为通过 D-H 理论获得的运动学参数。D-H 理论为刻画机器人运动学模型的标准理论工具，由 Denavit 与 Hartenberg 两人在 1955 年提出。D-H 理论引入了一个标准化框架，在该框架下，不同的机械臂可采用一套通用的 α , θ , a , d 参数描述运动学模型。其中，将第 k 个关节所对应的 D-H 参数代入公式(19)，就可以得到从 $k-1$ 到 k 个局部坐标系的转换矩阵的具体形式。矩阵 T_{k-1}^k 也符合公式(18)所定义的通用形式：左上 3×3 的矩阵用来表示第 $k-1$ 个坐标系与第 k 个坐标系的旋转关系，最右一列的 3×1 向量表示两者的平移关系，最末一行由常数 0 和 1 构成。

应用 D-H 理论建立机械臂运动学坐标系的步骤如下：

- 1) 从底座到末端，将每个关节依次记为 1, 2, ..., n;
- 2) 按照右手定则，将第 0 个坐标系分配至机器人底座，设定 z_0 ，将其对齐于关节 1 的旋转轴；
- 3) 设置 $k=1$ ，按如下步骤进行：
 - 设定 z_k ，将其对齐于关节 $k+1$ 的旋转轴，
 - 将 z_k 与 z_{k-1} 轴的交点设为第 k 个坐标系的原点；如果两轴不相交，选择同时垂直于 z_k 和 z_{k-1} 的轴线，将 z_k 与该线交点设为第 k 个坐标系的原点；
 - 设定 x_k 为同时垂直于 z_k 和 z_{k-1} 的轴线；如果 z_k 与 z_{k-1} 平行，选择远离 z_{k-1} 的方向为 x_k ；
 - 设定 y_k ，以组成右手坐标系
- 4) $k=k+1$ ，按照步骤(3)继续，直到 $k=n$ ；
- 5) 设定第 n 个坐标系为末端执行器坐标系，如图 12 所示：

z_n — approaching vector

y_n — sliding vector

x_n — normal vector

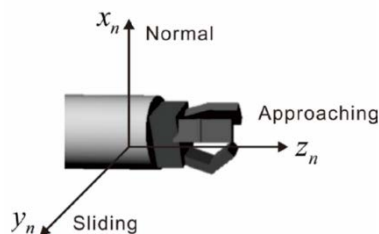


图 12. 末端执行器所对应的局部坐标系

运动学坐标系建立完成之后，D-H 参数定义为：

- θ_k 为 x_{k-1} 与 x_k 轴沿着 z_{k-1} 方向的夹角；
- d_k 为 x_{k-1} 与 x_k 轴沿着 z_{k-1} 方向的距离；
- a_k 为 z_{k-1} 与 z_k 轴沿着 x_k 方向的距离；
- α_k 为 z_{k-1} 与 z_k 轴沿着 x_k 方向的夹角。

考虑如图 13 所示的四自由度机械臂：

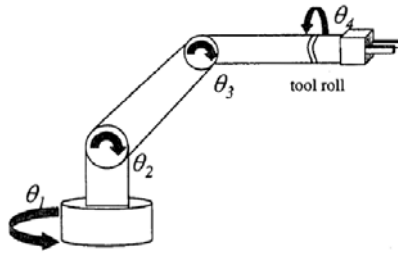


图 13. 四自由度机械臂

按照 D-H 理论建立的坐标系如图 14 所示：

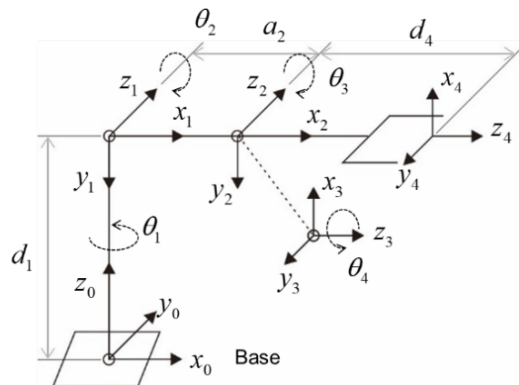


图 14. 四自由度机械臂的 D-H 坐标系

图 14 所对应的 D-H 参数值总结如下：

表 1. 四自由度机械臂对应的 D-H 参数

Axis	θ	d	a	α
1	θ_1	d_1	0	$-\pi/2$
2	θ_2	0	a_2	0
3	θ_3	0	0	$-\pi/2$
3	θ_4	d_4	0	0

基于 D-H 理论建立的坐标系不唯一。比如图 14 的坐标系中，z 轴可以设置为相反方向。

考虑图 15 所示的四自由度 SCARA 机械臂：

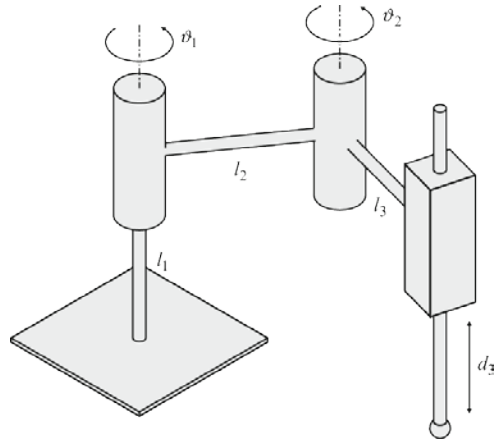


图 15. SCARA 机械臂 (DOI: 10.1007/978-90-481-3776-3_2)

对于 SCARA 机械臂的四个自由度，前两个自由度的旋转轴平行，两个关节在同一个平面内旋转，第三个自由度在垂直平面的方向上下移动，最后一个自由度的旋转轴也平行于前两个自由度。按照 D-H 理论建立的 SCARA 机械臂坐标系如图 16 所示：

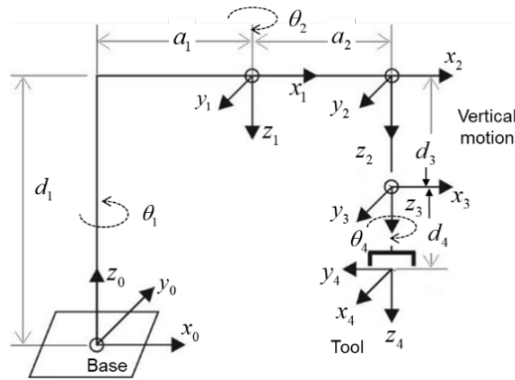


图 16. SCARA 机械臂的 D-H 坐标系

图 16 所对应的 D-H 参数值总结如下：

表 2. SCARA 机械臂的 D-H 参数

Axis	θ	d	a	α
1	θ_1	d_1	a_1	π
2	θ_2	0	a_2	0
3	0	d_3	0	0
3	θ_4	d_4	0	0

将以上表格的每行参数值代入公式(19)，可以得到一系列的转换矩阵： T_0^1 ， T_1^2 ， \dots ， T_{n-1}^n 。得到这一系列的转换矩阵之后，将其依次相乘： $T_0^1 \times T_1^2 \times \dots \times T_{n-1}^n = T_0^n$ ，即可得到末端执行器所对应的局部坐标系相对于底座的相对关系。

比如，对于表 2 中的 SCARA 机械臂 D-H 参数值，计算出从底座到末端执行机构的转换矩阵为：

$$T_{base}^{tool} = \begin{bmatrix} C_{124} & S_{124} & 0 & a_1 C_1 + a_2 C_{12} \\ S_{124} & -C_{124} & 0 & a_1 S_1 + a_2 S_{12} \\ 0 & 0 & -1 & d_1 - d_3 - d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (20)$$

考虑图 17 所示的两自由度平面机械臂， l_1 与 l_2 为两个连杆长度， q_1 与 q_2 为机械臂连杆轴线夹角， r_1 与 r_2 为末端执行器相对于原点的坐标：

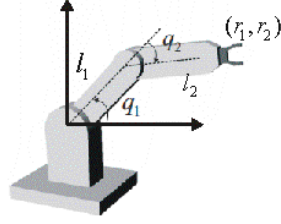


图 17. 两自由度平面机械臂

基于 D-H 理论计算从底座到末端执行机构的转换矩阵为：

$$T_{base}^{tool} = \begin{bmatrix} \cos(q_1 + q_2) & \sin(q_1 + q_2) & l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) \\ \sin(q_1 + q_2) & -\cos(q_1 + q_2) & l_1 \sin(q_1) + l_2 \sin(q_1 + q_2) \\ 0 & 0 & 1 \end{bmatrix} \quad (21)$$

有别于公式(20)描述的 4×4 齐次矩阵，公式(21)定义了一个降维的 3×3 齐次转换矩阵，原因在于所对应的机械臂仅具有在平面运动的两自由度。

如果仅考虑末端执行器相对于底座的位移，提取出公式(21)最末一列可得：

$$\begin{cases} r_1 = l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) \\ r_2 = l_1 \sin(q_1) + l_2 \sin(q_1 + q_2) \end{cases} \quad (22)$$

公式(22)即刻画了机械臂关节角度与末端执行机构位置的关系。如果采取通用表达形式，公式(22)可以表达为：

$$r = h(q) \quad (23)$$

其中 h 为非线性函数。

对(22)两边相对时间求导，可以得到末端速度与关节转速之间的关系：

$$\begin{cases} \dot{r}_1 = -l_1 \dot{q}_1 \sin(q_1) - l_2 (\dot{q}_1 + \dot{q}_2) \sin(q_1 + q_2) \\ \dot{r}_2 = l_1 \dot{q}_1 \cos(q_1) + l_2 (\dot{q}_1 + \dot{q}_2) \cos(q_1 + q_2) \end{cases} \quad (24)$$

整理成矩阵表达形式可得：

$$\begin{bmatrix} \dot{r}_1 \\ \dot{r}_2 \end{bmatrix} = \begin{bmatrix} -l_1 \sin(q_1) - l_2 \sin(q_1 + q_2) & -l_2 \sin(q_1 + q_2) \\ l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) & l_2 \cos(q_1 + q_2) \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \quad (25)$$

其中，等号右边的 2×2 矩阵为联系两个速度向量的 Jacobian 矩阵。如果写成通用表达形式可得：

$$\dot{r} = J(q)\dot{q} \quad (26)$$

末端执行器

机械臂通过末端的执行器执行具体任务。比如，图 18 左边的机械臂末端安装刮板以开展抹腻子任务，图 18 右边的机械臂末端安装吸盘以开展贴瓷片任务。在开展任务过程中，我们需要调整末端执行器的位置和姿态，使其适应期望任务。

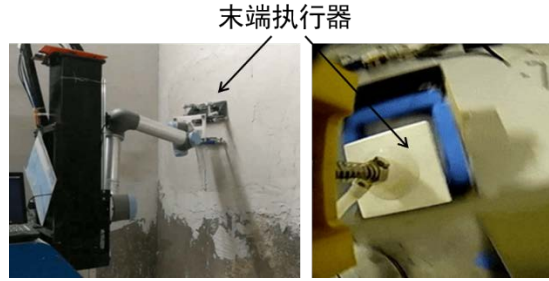


图 18. 各种不同的末端执行机构

基于 D-H 理论获得的齐次矩阵如公式(27)所示，刻画了从机器人底座到末端的转换矩阵；其中，左上的 3×3 旋转矩阵，表示末端所处局部坐标系与底座参考坐标系之间的旋转关系，右侧最末一列的 3×1 向量，表示两个坐标系之间的平移关系：

$$T_{base}^{tool} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & p_1 \\ R_{21} & R_{22} & R_{23} & p_2 \\ R_{31} & R_{32} & R_{33} & p_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (27)$$

基于公式(27)，如果将平移向量中的 p_1, p_2, p_3 元素提取出来，即可刻画机器人末端特征点相对于底座参考点的位移量。

类似地，如果将左上 3×3 矩阵提取出来，拼成 9×1 的列向量，可以刻画末端执行器相对于底座的旋转关系；然而，这种方式虽然直接，但表达方式过于冗余，影响机器人的实时操作性能；考虑到旋转关系可采用三个角度进行表示，构造一个 3×1 向量的表示方法将更为简洁。

我们考虑公式(27)左上部分旋转矩阵的最后一列： $r_3 = [R_{13}, R_{23}, R_{33}]$ ；该向量表示了 r_3 单位向量在 F 坐标系（即机械臂的固定底座）三轴方向上的投影；根

据图 12 可得， r_3 单位向量一般设置为末端执行器的旋转轴；因此，我们可以通过该向量表示末端执行器相对于底座的旋转关系，如图 19 所示。

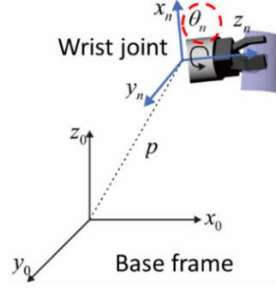


图 19. 机器人末端执行器相对于固定底座的姿态表示方法

虽然 r_3 向量确定了末端执行器相对于固定底座的旋转角度，它的具体姿态还必须包含该向量围绕 r_3 旋转的角度信息。该角度正对应于机械臂最后一个自由度的旋转角度（相当于机械臂的手腕位置），即 θ_n 。

因为，我们可以采用“前进方向（即 r_3 向量）+沿前进方向的旋转角度（即 θ_n ）”以完整地刻画末端执行器的姿态信息。在具体表示方法上，需要将两类信息组合起来；其中一个方法，是将旋转角度编码到 r_3 向量中：

$$w = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \end{bmatrix} = \begin{bmatrix} p \\ r_3 e^{\frac{\theta_n}{\pi}} \end{bmatrix} \quad (28)$$

公式(28)中 w 向量前三维表示末端执行器的位置，后三维表示末端执行器编码后的姿态信息。这里采用了指数函数进行编码，优点包括：不论 θ_n 为多少，指数函数的输出值总是正数，不会影响 r_3 向量表示的方向；指数函数具有单调变化性，可以通过求解指数函数的反函数反算 θ_n 。

考虑图 15 的 SCARA 机械臂，通过引入一个关节空间向量来表示机械臂的四个自由度：

$$q = [q_1, q_2, q_3, q_4]^T = [\theta_1, \theta_2, d_3, \theta_4]^T \quad (29)$$

其中， q_1, q_2, q_4 表示旋转自由度， q_3 表示平移自由度。基于 D-H 理论，已经建立了从底座到末端坐标系的转换矩阵，即公式(27)。我们按照公式(27)的形式，提取出最右列的 3×1 列向量，表示末端执行器的位置；然后提取 r_3 向量，加上末端转角编码，表示末端执行器的姿态；在这种表达方式中， q_4 即末端执行

器绕前进方向的旋转角度。以上刻画机械臂末端执行器的方式并不唯一。

$$w(q) = \begin{bmatrix} a_1 C_1 + a_2 C_{12} \\ a_1 S_1 + a_2 S_{12} \\ d_1 - q_3 - d_4 \\ 0 \\ 0 \\ -e^{\frac{q_4}{\pi}} \end{bmatrix} \quad (30)$$

逆运动学

机器人正向运动学是指从关节转角推得末端执行器的位置与姿态；顾名思义，机器人逆运动学，指给定末端执行器的期望位置，求解机器人的对应关节转角。因此，图 20 表示的逆运动学模块的输入与输出分别为末端执行器的位置与对应关节角度。

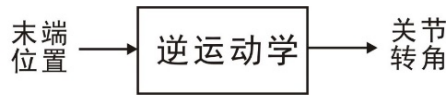


图 20. 逆运动学模块

求解机器人逆运动学的目的，是因为在任务空间定义机器人要执行任务的期望位置更方便与直观。比如说，对于图 21 所示的贴瓷片任务，机器人可以通过在墙上找到已经贴好的瓷片，迅速设置末端执行器的期望位置，即，已贴好瓷片的相邻位置。

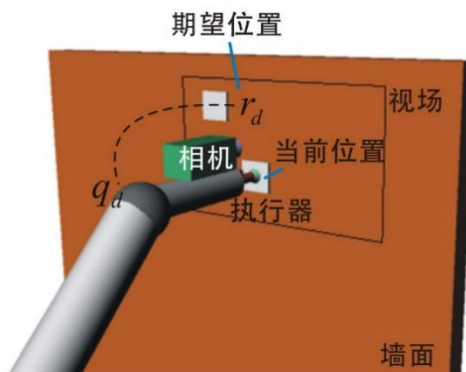


图 21. 墙面的贴瓷片任务

因此，贴瓷片任务中的期望位置可以在任务空间（即墙面上）以非常直观的形式迅速获取，但是其对应的关节转角需要通过求解逆运动学才能得到。

考虑公式(4)描述的单关节机械臂，给定机械臂末端在 2D 平面的位置：\$x_0\$，\$y_0\$，其逆运动学对应的关节角度可通过求解(4)中任意一个方程获得。比如，

$$\theta = \sin^{-1} \left(\frac{y_0}{l} \right) \quad (31)$$

考虑图 15 表示的 SCARA 机械臂，其逆运动学问题可以描述为：给定公式 (30) 中的 $w(q)$ 向量，求解 q_1, q_2, q_3, q_4 具体取值。其中一种求解方法步骤如下：

- 1) 通过观察，发现 $w(q)$ 向量前两个元素的平方和，等于 $a_1^2 + 2a_1a_2C_2 + a_2^2$ ：

$$w_1^2 + w_2^2 = a_1^2 + 2a_1a_2C_2 + a_2^2 \quad (32)$$

通过公式(32)可以反解得出第二个关节角度：

$$q_2 = \pm \cos^{-1} \frac{w_1^2 + w_2^2 - a_1^2 - a_2^2}{2a_1a_2}$$

- 2) 将求解得到的 q_2 代入方程(30)，可得

$$\begin{cases} w_1 = (a_1 + a_2C_2)C_1 + a_2S_2S_1 \\ w_2 = (a_1 + a_2C_2)S_1 - a_2S_2C_1 \end{cases} \quad (33)$$

将 q_2 视为已知量，将方程组(33)经过变换可得：

$$\begin{cases} S_1 = \frac{a_2S_2w_1 + (a_1 + a_2C_2)w_2}{(a_2S_2)^2 + (a_1 + a_2C_2)^2} \\ C_1 = \frac{(a_1 + a_2C_2)w_1 - a_2S_2w_2}{(a_2S_2)^2 + (a_1 + a_2C_2)^2} \end{cases} \quad (34)$$

以上方程两者相除，得到 q_1 的正切值，对得到的结果求解反三角函数，得到了 q_1 的角度值：

$$q_1 = \tan^{-1} \frac{a_2S_2w_1 + (a_1 + a_2C_2)w_2}{(a_1 + a_2C_2)w_1 - a_2S_2w_2} \quad (35)$$

- 3) 第三个关节（即唯一的平移自由度）可以通过向量 $w(q)$ 的第三个元素值求得，即：

$$q_3 = d_1 - w_3 - d_4 \quad (36)$$

- 4) 第四个关节的取值可通过对编码后的指数函数求逆获得：

$$q_4 = \pi \ln(-w_6) \quad (37)$$

以上方法一般称为解析法，指通过观察前向运动学方程的结构，依次消去其它变量达到的求解。解析法没有固定的流程，不同的机械臂有不同的解法。随着自由度数量增多，解析法的求解难度迅速增加；存在未知的机械臂运动学参数情况下，无法求解出末端期望位置与姿态所对应的精准角度。

除了解析法，也可以采用数值法求解机器人逆运动学。数值法引入优化问题，

通过最小化末端执行器向量当前值与期望值的差值，求解对应的关节角度：

$$q = \operatorname{argmin} ||w(q) - w_d|| \quad (38)$$

因此可以用求解优化问题的方法进行计算。此外，也可以用启发式算法来求解公式(38)表示的优化问题。或者，基于人工神经网络，通过采集并标注大量末端执行器与对应关节角度的样本，进行神经网络训练，训练成熟的网络即可刻画机器人逆运动学模型。

冗余自由度

公式(26)中的 Jacobian 矩阵用来描述关节转速与末端速度的关系。该矩阵的大小为 $m \times n$ ；其中， m 为任务空间的维度， n 为关节自由度数量；如果 $m=n$ ，关节空间和任务空间的维度一致，Jacobian 矩阵为方阵。

如果 $m < n$ ，关节空间的维度大于任务空间的维度，机械臂具有冗余自由度；对于任务空间给定任务，机械臂能够通过多种方式去实现；因此，除了完成末端执行器的主任务之外，机械臂能够通过额外的自由度去完成其他任务（如，改变整体外形以进行避障）。如果 $m > n$ ，机械臂处于欠驱动状态，需要通过更少的自由度控制更高维的任务。

机械臂是否具有冗余自由度取决于它所要执行的任务。考虑图 22 所示的四自由度平面机器人，每个自由度的旋转轴均垂直于纸面；对于末端执行器要完成的位置控制，机器人具有冗余自由度。

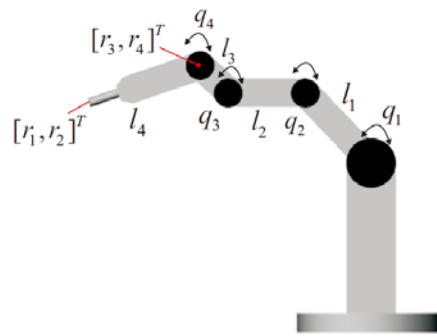


图 22. 四自由度平面机器人

具体来说，该机器人的末端位置可以基于前向运动学模型计算得到：

$$r = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} l_1 c_1 + l_2 c_{12} + l_3 c_{123} + l_4 c_{1234} \\ l_1 s_1 + l_2 s_{12} + l_3 s_{123} + l_4 s_{1234} \end{bmatrix} \in \mathbb{R}^2 \quad (39)$$

对公式(39)两边相对于时间求导数，整理后可推得该机械臂的 Jacobian 矩阵为：

$$J(q) = \begin{bmatrix} -(l_1 s_1 + l_2 s_{12} + l_3 s_{123} + l_4 s_{1234}) & -(l_2 s_{12} + l_3 s_{123} + l_4 s_{1234}) & -(l_3 s_{123} + l_4 s_{1234}) & -l_4 s_{1234} \\ l_1 c_1 + l_2 c_{12} + l_3 c_{123} + l_4 c_{1234} & l_2 c_{12} + l_3 c_{123} + l_4 c_{1234} & l_3 c_{123} + l_4 c_{1234} & l_4 c_{1234} \end{bmatrix} \quad (40)$$

该矩阵的维度为 2×4 ，其中，行数对应于任务空间的维数，列数对应了关节空间的维数。

如果除了末端执行器之外，我们再增加一个腕部的特征点，新加特征点与关节转角的关系可推导为：

$$\begin{bmatrix} r_3 \\ r_4 \end{bmatrix} = \begin{bmatrix} l_1 c_1 + l_2 c_{12} + l_3 c_{123} \\ l_1 s_1 + l_2 s_{12} + l_3 s_{123} \end{bmatrix} \quad (41)$$

其中， r_3 与 r_4 是新加特征点在笛卡尔空间的位置特征。然后，将任务空间向量扩展为两个特征点的位置（即，点特征变为线特征），可得：

$$r = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{bmatrix} \quad (42)$$

然后，方程(42)两边相对时间进行求导，再整理成向量与矩阵的形式，就可以得到下面这个新的 4×4 矩阵，

$$\dot{r} = \begin{bmatrix} \dot{r}_1 \\ \dot{r}_2 \\ \dot{r}_3 \\ \dot{r}_4 \end{bmatrix} = J(q)\dot{q} = \begin{bmatrix} -(l_1 s_1 + l_2 s_{12} + l_3 s_{123} + l_4 s_{1234}) & -(l_2 s_{12} + l_3 s_{123} + l_4 s_{1234}) & -(l_3 s_{123} + l_4 s_{1234}) & -l_4 s_{1234} \\ l_1 c_1 + l_2 c_{12} + l_3 c_{123} + l_4 c_{1234} & l_2 c_{12} + l_3 c_{123} + l_4 c_{1234} & l_3 c_{123} + l_4 c_{1234} & l_4 c_{1234} \\ -(l_1 s_1 + l_2 s_{12} + l_3 s_{123}) & -(l_2 s_{12} + l_3 s_{123}) & -l_3 s_{123} & 0 \\ l_1 c_1 + l_2 c_{12} + l_3 c_{123} & l_2 c_{12} + l_3 c_{123} & l_3 c_{123} & 0 \end{bmatrix} \times \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} \quad (43)$$

改进之后的 Jacobian 矩阵为 4×4 ，对于由 r_3 ， r_4 共同构成的“线特征”不具备冗余性，可以唯一确定期望任务的关节角度。

奇异性

当 Jacobian 矩阵满秩且为方阵, 可以通过 Jacobian 的逆矩阵反推出末端速度所对应的关节转速:

$$\dot{q} = J^{-1}(q)\dot{r} \quad (44)$$

对于冗余机器人, Jacobian 矩阵不是方阵, 通过引入广义矩阵来表示求逆的关系:

$$J^+(q) = J^T(q) \left(J(q)J^T(q) \right)^{-1} \quad (45)$$

$\begin{matrix} n \times m & n \times m & m \times m \end{matrix}$

公式(45)中的第二项为方阵求逆运算; 若该方阵可逆, 求逆结果再左乘 Jacobian 矩阵的转置矩阵, 即可获得广义逆阵。

根据公式(45)的定义, 该广义逆阵左乘 Jacobian 矩阵可以得到单位矩阵:

$$J(q)J^+(q) = I \quad (46)$$

机械臂的奇异性发生在某些导致 Jacobian 的逆矩阵或者广义逆阵不存在的某些角度配置; 在这些情况下, 机器人的控制输入将变得无穷大; 任务空间的微小运动需要关节无穷大的转速才能实现, 可能导致机器人产生一些危险的行为。

机械臂的关节奇异位置对应于 Jacobian 矩阵非满秩的关节角度。对于非冗余机械臂, 该角度值对应于 Jacobian 矩阵行列式为零的解; 对于冗余机械臂, 该角度值可以通过求解 Jacobian 矩阵与它转置乘积的行列式为零来得到。

一个机械臂系统的奇异位置可以分为外部奇异与内部奇异位置。外部奇异位置发生在机械臂工作空间的外部边界; 内部奇异位置可进一步划分为边界奇异与内部奇异两种。

考虑图 23 所示的两自由度平面机械臂, 其对应的 Jacobian 矩阵如公式(25)所示。该机械臂的关节奇异位置可通过求解 Jacobian 矩阵的行列式为零来获得:

$$\det[J(q)] = l_1 l_2 \sin(q_2) = 0 \quad (47)$$

方程(47)的解有两个: $q_2=0$ 与 $q_2=180^\circ$ 。这两个角度分别对应于: 第二个关节完全展开的时候(外边界奇异位置), 或者是它完全收回来的时候(内边界奇异位置); 当机械臂运动到这两个位置的时候, Jacobian 矩阵的逆矩阵或者广义逆阵不存在, 导致机械臂发生了奇异性; 如果机械臂的控制输入中包含有逆矩阵或者广义逆矩阵的信息, 应该设计机械臂末端的期望位置或轨迹使其避开奇异位置。

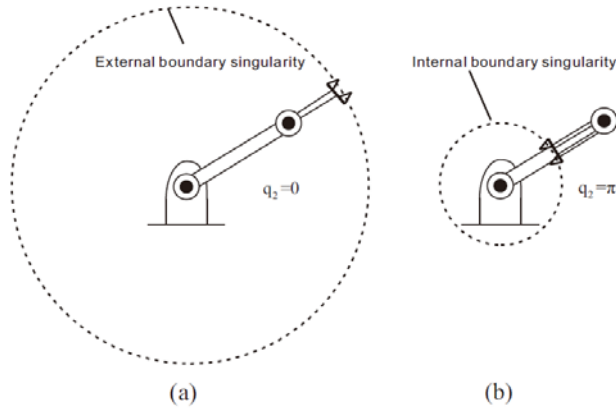


图 23. 两自由度机械臂的奇异位置：(a) $q_2=0$ （外边界奇异）(b) $q_2=180^\circ$ （内边界奇异）

除了从算法层面进行限制，也可从机械结构上加以限制，使得图 23 所示的机械臂无法完全展开或收拢，从而达到避免奇异性的目的。

再考虑图 24 所示的三自由度机械臂，它的第一个自由度旋转轴垂直于底座，第二与第三个自由度的旋转轴垂直于第一个自由度的旋转轴。因此，机器人具有在 3D 空间的运动能力。

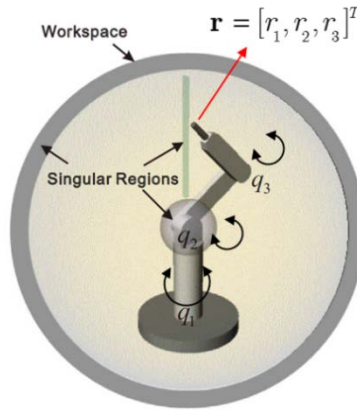


图 24. 三自由度的机械臂

采用向量 \mathbf{r} 表示末端执行器的位置，基于机械臂的前向运动学模型可以建立末端执行器位置与关节转角的关系：

$$\mathbf{r} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} = \begin{bmatrix} \cos(q_1)[l_2 \cos(q_2) + l_3 \cos(q_2 + q_3)] \\ \sin(q_1)[l_2 \cos(q_2) + l_3 \cos(q_2 + q_3)] \\ l_1 - l_2 \sin(q_2) - l_3 \sin(q_2 + q_3) \end{bmatrix} \in \mathbb{R}^3 \quad (48)$$

所对应的 Jacobian 矩阵为：

$$\mathbf{J}(\mathbf{q}) = \begin{bmatrix} -s_1(l_2 c_2 + l_3 c_{23}) & -s_2 c_1 l_2 - s_{23} l_3 c_1 & -s_{23} l_3 c_1 \\ -c_1(l_2 c_2 + l_3 c_{23}) & -s_2 s_1 l_2 - s_{23} l_3 s_1 & -s_{23} l_3 s_1 \\ 0 & -c_2 l_2 - c_{23} l_3 & -c_{23} l_3 \end{bmatrix} \in \mathbb{R}^{3 \times 3} \quad (49)$$

类似地，机械臂的奇异位置可以通过求解 Jacobian 行列式等于零获得，即：

$$\det[J(q)] = [l_2 \cos(q_2) + l_3 \cos(q_2 + q_3)] l_3 l_2 \sin(q_3) = 0 \quad (50)$$

方程的解包括： $q_3=0$ ， $q_3=\pi$ ， $l_2 \cos(q_2) + l_3 \cos(q_2 + q_3) = 0$ 。从图 24 中可以看到， $q_3=0$ 对应于机械臂的第三自由度完全展开的情况； $q_3 = \pi$ ，对应于第三自由度完全收回来；第三个奇异位置对应于图 24 中的长条形区域。类似地，如果该机械臂的控制输入中包含有 Jacobian 的逆矩阵或者广义逆矩阵，所设计的期望位置应该避开这些奇异位置。

除了由 Jacobian 矩阵引起的奇异性，末端执行器姿态表征方式中也存在奇异性问题。为了刻画末端执行器的姿态，可以采用 yaw, pitch, roll 三个角度刻画执行器绕三轴旋转的情况，通过以下的旋转矩阵公式来表达：

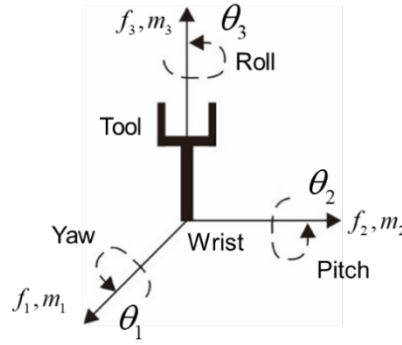


图 25. 末端执行器

$$YPR(\theta) = R_3(\theta_3)R_2(\theta_2)R_1(\theta_1) = \begin{bmatrix} C_2 C_3 & S_1 S_2 C_3 - C_1 S_3 & C_1 S_2 C_3 + S_1 S_3 \\ C_2 S_3 & S_1 S_2 S_3 + C_1 C_3 & C_1 S_2 S_3 - S_1 C_3 \\ -S_2 & S_1 C_2 & C_1 C_2 \end{bmatrix} \quad (51)$$

给定公式(51)中矩阵元素的具体值，可以通过公式(51)反推出三轴旋转角度，用来表示末端执行器的姿态：

$$\begin{cases} \theta_2 = \sin^{-1}(-YPR_{31}) \\ \theta_1 = \sin^{-1}\left(\frac{YPR_{32}}{\cos \theta_2}\right) \\ \theta_3 = \cos^{-1}\left(\frac{YPR_{11}}{\cos \theta_2}\right) \end{cases} \quad (52)$$

通过观察可发现，这种表达方式存在奇异性，奇异性发生在：

$$\cos \theta_2 = 0 \quad (53)$$

这种奇异性有别于 Jacobian 矩阵的奇异性，造成这种奇异性的原因在于表达方式本身存在的局限：如果采用三个旋转角度表示末端执行器相对于三轴的旋转，在通过矩阵元素求解旋转角度的过程中，出现了某些角度值无法求解的情况。

为了解决表达方式带来的奇异性，我们可以在表达方式中引入一定程度的冗余性（比如四元数表示法）。

零空间

具有冗余自由度的机械臂能够执行末端执行器之外的次级任务，如图 26 所示的七自由度 KUKA iiwa 机械臂。对于一个具有冗余自由度的机械臂，对应的 Jacobian 矩阵的行数（即任务空间的维数）少于列数（即关节自由度数量）。其相应的零矩阵定义为：

$$N(q) = I_n - J^+(q)J(q) \quad (54)$$



图 26. 七自由度 KUKA iiwa 机械臂

末端执行器的零空间是指：当机械臂关节速度位于该零空间中，末端执行器的运动速度保持为零；基于零空间矩阵 $N(q)$ 对某些关节转速进行投影运算，计算得到的关节转速，将不会改变机械臂末端的速度。

零空间矩阵的一些运算性质如下：

$$J(q)N(q) = J(q)(I_n - J^+(q)J(q)) = 0 \quad (55)$$

$$N(q)J^+(q) = 0 \quad (56)$$

$$N^2(q) = N(q) \quad (57)$$

以上性质均可根据零空间矩阵在公式(54)中的定义推得。

零空间矩阵适用于具有冗余自由度的机械臂；如果 $m=n$ ，机械臂不具有冗余自由度，广义矩阵将退化为逆矩阵 $J^{-1}(q)$ ；根据零空间矩阵定义， $N(q)$ 恒等于零。

零空间矩阵可以在不影响末端执行器主任务的前提下，驱动冗余自由度执行次级任务（比如避障）。相关的控制器输入可以定义为：

$$u = J^+(q)(\dot{r}_c - K_c(r - r_c)) + N(q)(\dot{q}_d - K_p(q - q_d)) \quad (58)$$

其中， r_c 为末端执行器在任务空间的期望位置，公式(58)中第二项定义了冗余关节需要执行的任务： q_d 为冗余关节对应的期望角度， K_c, K_p 为控制参数。

将控制输入代入机械臂运动学方程：

$$\dot{q} = u \quad (59)$$

得到闭环方程为：

$$\dot{q} = J^+(q)(\dot{r}_c - K_c(r - r_c)) + N(q)(\dot{q}_d - K_p(q - q_d)) \quad (60)$$

为了对公式(60)描述的闭环系统进行稳定性分析，对公式(60)左右两边分别乘以 $J(q)$ 与 $N(q)$ ，可得：

$$\begin{cases} \dot{r} = \dot{r}_c - K_c(r - r_c) \\ N(q)((\dot{q} - \dot{q}_d) + K_p(q - q_d)) = 0 \end{cases} \quad (61)$$

公式(61)说明：在稳态的时候，机械臂末端执行器的位置停留在期望位置；同时，冗余关节角度趋于期望角度，完成了次级任务（比如，避障任务可以通过定义期望关节转角 q_d 来实现）。

视觉伺服

之前提到机械臂的 **Jacobian** 矩阵用来描述关节转速与任务空间速度的关系。该任务空间为笛卡尔坐标系，末端执行器在笛卡尔坐标系的位置基于前向运动学模型求解。

除了笛卡尔坐标系，我们也可以将末端执行器的位置定义在传感反馈空间，即采用外部传感器直接测量末端执行器的位置：

$$x = [x_1, x_2, \dots, x_p]^T \in \mathbb{R}^p \quad (62)$$

公式(62)定义了传感反馈空间向量 x 。

通过将末端执行器表征在传感反馈空间，可以引入另外一个 **Jacobian** 矩阵 $J_s(r)$ ，表示从笛卡尔坐标系到传感反馈空间的映射

$$\dot{x} = J_s(r)\dot{r} \quad (63)$$

基于公式(26)，可以进一步将传感反馈空间向量与关节空间向量联系起来：

$$\dot{x} = J_s(r)J_m(q)\dot{q} = J(q)\dot{q} \quad (64)$$

其中, $J(q)$ 表示机械臂的整体 Jacobian 矩阵, 表征了从关节空间到传感反馈空间的映射, 即 $J_m(q)$ 与 $J_s(r)$ 两个矩阵的乘积; 矩阵 $J_m(q)$ 表示从关节空间到笛卡尔坐标系的映射, 矩阵 $J_s(r)$ 表示从笛卡尔坐标系到传感反馈空间的映射。

整体的 Jacobian 矩阵可以参数化表示成回归矩阵与参数向量的乘积:

$$\dot{x} = J(q)\dot{q} = Y_k(q, \dot{q})\theta_k \quad (65)$$

其中, Y_k 表示回归矩阵, 该矩阵仅包含状态变量, 状态变量的取值可通过传感器直接测量, 或者对测量结果进行数值计算获得; 向量 θ_k 包含机械臂运动学模型参数与传感器模型参数, 比如, 机械臂末端所抓取工具的长度、角度等。在某些情况下 (如机械臂抓取未校正工具进行操作), 我们无法获取这些参数的精准值。为了解决参数不确定性问题, 可以基于这条性质, 设计一些自适应更新律, 在线更新未知的运动学参数。

机械臂常用的传感器之一为相机。基于相机的视觉反馈具有较多优点: 可以捕获丰富的环境信息; 传感器价格相对便宜; 具有较高的鲁棒性。如果采用相机作为外部传感器, 并将机械臂末端执行器定义为特征, 特征位置可以通过图像空间的像素值进行描述:

$$x = [u, v]^T \in \mathbb{R}^2 \quad (66)$$

即传感空间向量 x 为 2×1 的向量, u, v 分别表示图像空间的横纵坐标。采用视觉反馈引导机械臂进行操作任务的系统一般称为视觉伺服系统 (visual servoing)。视觉伺服系统根据不同的相机成像几何学具有不同的运动学模型。其中, 常见的相机模型之一为小孔成像模型: 将一个点投影到图像平面的过程建模为光线通过一个透镜的中心, 如图 27 所示。

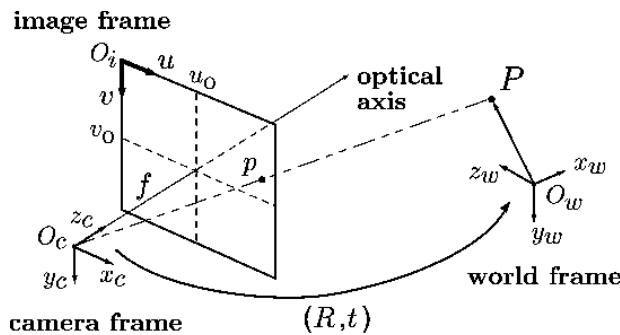


图 27. 小孔成像相机模型

在确定相机模型的前提下, 图像 Jacobian 矩阵可以用来描述末端执行器的运

动速度与图像空间像素变化的关系。考虑图 27 表示的单孔相机模型，其所对应的图像 Jacobian 矩阵为：

$$J_s(r) = \begin{bmatrix} \frac{f}{z} & 0 & -\frac{u}{z} & -\frac{uv}{f} & \frac{f^2+u^2}{f} & -v \\ 0 & \frac{f}{z} & -\frac{v}{z} & -\frac{f^2+v^2}{f} & \frac{uv}{f} & u \end{bmatrix} \quad (67)$$

公式(67)描述了一个 2×6 的图像 Jacobian 矩阵，矩阵的两行元素对应于图像中的一个像素点，6 列元素分别对应于末端的三个平移与三个旋转速度， f 表示相机的焦距。

图像 Jacobian 矩阵的维度将根据机械臂的自由度数量以及具体的相机配置发生变化。比如，如果末端执行器仅有平移运动，没有旋转运动，可以只考虑公式(67)表示的矩阵的前三列元素，将矩阵简化为：

$$J_s(r) = \begin{bmatrix} \frac{f}{z} & 0 & -\frac{u}{z} \\ 0 & \frac{f}{z} & -\frac{v}{z} \end{bmatrix} \quad (68)$$

进一步地，如果末端执行器仅在平面运动，且该平面平行于相机的成像平面，图像 Jacobian 可进一步简化为公式(69)表达的 2×2 的矩阵；该矩阵仅包含距离与焦距信息，元素为常数值，不随时间发生变化。

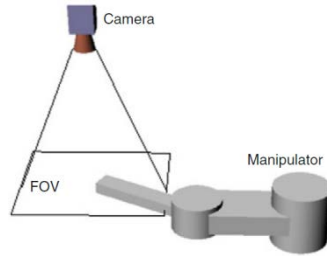


图 28. 相机成像平面平行于平面运动机械臂

$$J_s(r) = \begin{bmatrix} \frac{f}{z} & 0 \\ 0 & \frac{f}{z} \end{bmatrix} \quad (69)$$

该矩阵仅对特征在笛卡尔空间的位置进行“缩放”运算，不涉及平移与旋转。

对于图 28 所示的视觉引导机械臂，将公式(69)表达的图像 Jacobian 矩阵与机械臂的 Jacobian 矩阵相乘，即可得关节空间到图像空间的整体 Jacobian 矩阵：

$$J(q) = J_s J_m(q) = \begin{bmatrix} -\frac{f}{z}(l_1 s_1 + l_2 s_{12}) & -\frac{f}{z} l_2 s_{12} \\ \frac{f}{z}(l_1 c_1 + l_2 c_{12}) & \frac{f}{z} l_2 c_{12} \end{bmatrix} \in \mathbb{R}^{2 \times 2} \quad (70)$$

该矩阵可以将关节转角速度映射到图像空间的像素运动速度：

$$\begin{aligned}\dot{x} = J_s J_m(q) \dot{q} &= \begin{bmatrix} \frac{f}{z} & 0 \\ 0 & \frac{f}{z} \end{bmatrix} \begin{bmatrix} -l_1 s_1 - l_2 s_{12} & -l_2 s_{12} \\ l_1 c_1 + l_2 c_{12} & l_2 c_{12} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \\ &= \begin{bmatrix} -\frac{f}{z} l_1 s_1 \dot{q}_1 - \frac{f}{z} l_2 s_{12} (\dot{q}_1 + \dot{q}_2) \\ \frac{f}{z} l_1 c_1 \dot{q}_1 + \frac{f}{z} l_2 c_{12} (\dot{q}_1 + \dot{q}_2) \end{bmatrix} \quad (71)\end{aligned}$$

公式(71)的右边可以通过参数化，表达成回归矩阵与运动学参数向量乘积的形式：

$$\dot{x} = \begin{bmatrix} -s_1 \dot{q}_1 & -s_{12} (\dot{q}_1 + \dot{q}_2) \\ c_1 \dot{q}_1 & c_{12} (\dot{q}_1 + \dot{q}_2) \end{bmatrix} \begin{bmatrix} \frac{f}{z} l_1 \\ \frac{f}{z} l_2 \end{bmatrix} = Y_k(q, \dot{q}) \theta_k \quad (72)$$

其中， Y_k 为 2×2 矩阵， θ_k 为 2×1 向量；假设相机没有校正或机械臂抓取了未知工具进行操作（比如 l_1 或者 l_2 未知），运动学参数向量的精确值也是未知的；可以通过公式(72)描述的性质，设计自适应方法在线更新向量值；在相机未校正或运动学参数未知的情况下，仍然可以通过视觉反馈，通过控制图像特征到期望位置去执行操作任务。