

第2章 机器人位置运动学

2.1 引言

本章将研究机器人正逆运动学。当已知所有的关节变量时，可用正运动学来确定机器人末端手的位姿。如果要使机器人末端手放在特定的点上并且具有特定的姿态，可用逆运动学来计算出每一关节变量的值。首先利用矩阵建立物体、位置、姿态以及运动的表示方法，然后研究直角坐标型、圆柱坐标型以及球坐标型等不同构型机器人的正逆运动学，最后利用 Denavit-Hartenberg(D-H)表示法来推导机器人所有可能构型的正逆运动学方程。

实际上，机器手型的机器人没有末端执行器，多数情况下，机器人上附有一个抓持器。根据实际应用，用户可为机器人附加不同的末端执行器。显然，末端执行器的大小和长度决定了机器人的末端位置，即如果末端执行器的长短不同，那么机器人的末端位置也不同。在这一章中，假设机器人的末端是一个平板面，如有必要可在其上附加末端执行器，以后便称该平板面为机器人的“手”或“端面”。如有必要，还可以将末端执行器的长度加到机器人的末端来确定末端执行器的位姿。

2.2 机器人机构

机器手型的机器人具有多个自由度 (DOF)，并有三维开环链式机构。

在具有单自由度的系统中，当变量设定为特定值时，机器人机构就完全确定了，所有其他变量也就随之而定。如图 2.1 所示的四杆机构，当曲柄转角设定为 120° 时，则连杆与摇杆的角度也就确定了。然而在一个多自由度机构中，必须独立设定所有的输入变量才能知道其余的参数。机器人就是这样的多自由度机构，必须知道每一关节变量才能知道机器人的手处在什么位置。

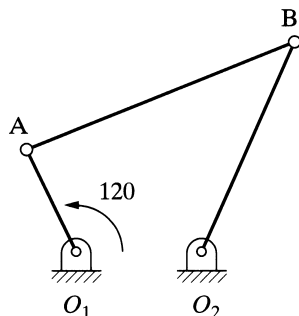


图 2.1 具有单自由度闭环的四杆机构

如果机器人要在空间运动，那么机器人就需要具有三维的结构。虽然也可能有二维多自由度的机器人，但它们并不常见。

机器人是开环机构，它与闭环机构不同（例如四杆机构），即使设定所有的关节变量，也不能确保机器人的手准确地处于给定的位置。这是因为如果关节或连杆有丝毫的偏差，该关节之后的所有关节的位置都会改变且没有反馈。例如，在图 2.2 所示的四杆机构中，如果连杆 AB 偏移，它将影响 O_2B 杆。而在开环系统中（例如机器人），由于没有反馈，之后的所有构件都会发生偏移。于是，在开环系统中，必须不断测量所有关节和连杆的参数，或者监控系统的末端，以便知道机器的运动位置。通过比较如下的两个连杆机构的向量方程，可以表示出这种差别，该向量方程表示了不同连杆之间的关系。

$$\overrightarrow{O_1A} + \overrightarrow{AB} = \overrightarrow{O_1O_2} + \overrightarrow{O_2B} \quad (2.1)$$

$$\overrightarrow{O_1A} + \overrightarrow{AB} + \overrightarrow{BC} = \overrightarrow{O_1C} \quad (2.2)$$

可见，如果连杆 AB 偏移，连杆 O_2B 也会相应地移动，式 (2.1) 的两边随连杆的变化而改变。而另一方面，如果机器人的连杆 AB 偏移，所有的后续连杆也会移动，除非 O_1C 有其他方法测量，否则这种变化是未知的。

为了弥补开环机器人的这一缺陷，机器人手的位置可由类似摄像机的装置来进行不断测量，于是机器人需借助外部手段（比如辅助手臂或激光束）来构成闭环系统。或者按照常规做法，也可通过增加机器人连杆和关节强度来减少偏移，采用这种方法将导致机器人重量重、体积大、动作慢，而且它的额定负载与实际负载相比非常小。

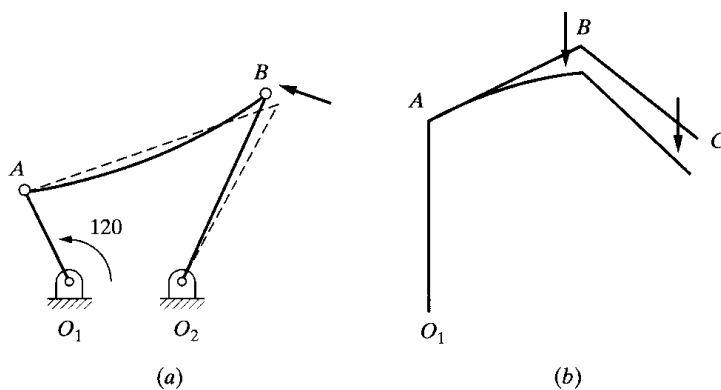


图 2.2 (a)闭环机构；(b)开环机构

2.3 机器人运动学的矩阵表示

矩阵可用来表示点、向量、坐标系、平移、旋转以及变换，还可以表示坐标系中的物体和其他运动元件。

2.3.1 空间点的表示

空间点 P（如图 2.3 所示）可以用它的相对于参考坐标系的三个坐标来表示：

$$P = a_x i + b_y j + c_z k \quad (2.3)$$

其中， a_x, b_y, c_z 是参考坐标系中表示该点的坐标。显然，也可以用其他坐标来表示空间点的位置。

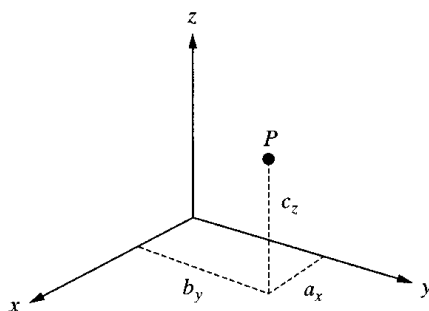


图 2.3 空间点的表示

2.3.2 空间向量的表示

向量可以由三个起始和终止的坐标来表示。如果一个向量起始于点 A，终止于点 B，那么它可以表示为 $\vec{P}_{AB} = (B_x - A_x)i + (B_y - A_y)j + (B_z - A_z)k$ 。特殊情况下，如果一个向量起始于原点（如图 2.4 所示），则有：

$$\overline{P} = a_x i + b_y j + c_z k \quad (2.4)$$

其中 a_x, b_y, c_z 是该向量在参考坐标系中的三个分量。实际上，前一节的点 P 就是用连接到该点的向量来表示的，具体地说，也就是用该向量的三个坐标来表示。

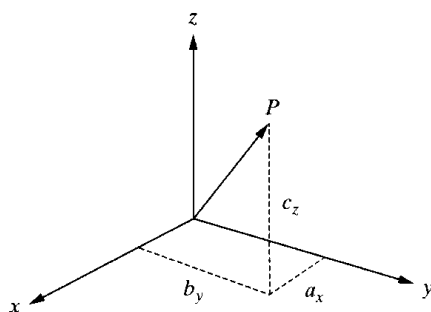


图 2.4 空间向量的表示

向量的三个分量也可以写成矩阵的形式，如式 (2.5) 所示。在本书中将用这种形式来表示运动分量：

$$\overline{P} = \begin{bmatrix} a_x \\ b_y \\ c_z \end{bmatrix} \quad (2.5)$$

这种表示法也可以稍做变化：加入一个比例因子 w ，如果 x, y, z 各除以 w ，则得到 a_x, b_y, c_z 。于是，这时向量可以写为：

$$\overline{P} = \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}, \text{ 其中 } a_x = \frac{x}{w}, b_y = \frac{y}{w}, \text{ 等等} \quad (2.6)$$

变量 w 可以为任意数，而且随着它的变化，向量的大小也会发生变化，这与在计算机图形学中缩放一张图片十分类似。随着 w 值的改变，向量的大小也相应地变化。如果 w 大于 1，向量的所有分量都变大；如果 w 小于 1，向量的所有分量都变小。这种方法也用于计算机图形学中改变图形与画片的大小。

如果 w 是 1，各分量的大小保持不变。但是，如果 $w=0$ ， a_x, b_y, c_z 则为无穷大。在这种情况下， x, y 和 z （以及 a_x, b_y, c_z ）表示一个长度为无穷大的向量，它的方向即为该向量所表示的方向。这就意味着方向向量可以由比例因子 $w=0$ 的向量来表示，这里向量的长度并不重要，而其方向由该向量的三个分量来表示。

例 2.1 有一个向量 $P=3i+5j+2k$ ，按如下要求将其表示成矩阵形式：

- (1) 比例因子为 2
- (2) 将它表示为方向的单位向量

解：

该向量可以表示为比例因子为 2 的矩阵形式，当比例因子为 0 时，则可以表示为方向向量，结果如下：

$$\overline{P} = \begin{bmatrix} 6 \\ 10 \\ 4 \\ 2 \end{bmatrix} \quad \text{和} \quad \overline{P} = \begin{bmatrix} 3 \\ 5 \\ 2 \\ 0 \end{bmatrix}$$

然而，为了将方向向量变为单位向量，须将该向量归一化使之长度等于 1。这样，向量的每一个分量都要除以三个分量平方和的开方：

$$\lambda = \sqrt{P_x^2 + P_y^2 + P_z^2} = 6.16, \text{ 其中 } P_x = \frac{3}{6.16} = 0.487, P_y = \frac{5}{6.16}, \text{ 等等和}$$

$$\overline{P}_{unit} = \begin{bmatrix} 0.487 \\ 0.811 \\ 0.324 \\ 0 \end{bmatrix}$$

2.3.3 坐标系在固定参考坐标系原点的表示

一个中心位于参考坐标系原点的坐标系由三个向量表示，通常着三个向量相互垂直，称为单位向量 $\overline{n}, \overline{o}, \overline{a}$ ，分别表示法线（normal）、指向（orientation）和接近（approach）向量（如图 2.5 所示）。正如 2.3.3 节所述，每一个单位向量都由它们所在参考坐标系着的三个分量表示。这样，坐标系 F 可以由三个向量以矩阵的形式表示为：

$$F = \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix} \quad (2.7)$$

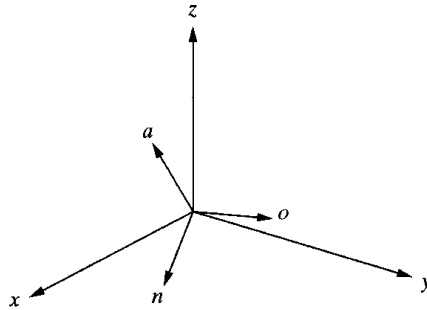


图 2.5 坐标系在参考坐标系原点的表示

2.3.4 坐标系在固定参考坐标系中的表示

如果一个坐标系不再固定参考坐标系的原点（实际上也可包括在原点的情况），那么该坐标系的原点相对于参考坐标系的位置也必须表示出来。为此，在该坐标系原点与参考坐标系原点之间做一个向量来表示该坐标系的位置（如图 2.6 所示）。这个向量由相对于参考坐标系的三个向量来表示。这样，这个坐标系就可以由三个表示方向的单位向量以及第四个位置向量来表示。

$$F = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

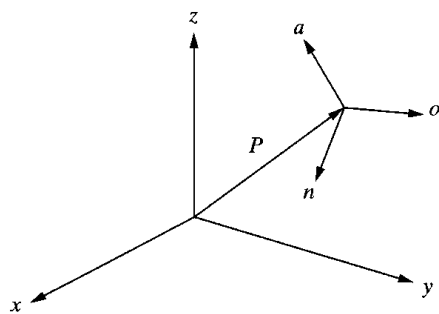


图 2.6 一个坐标系在另一个坐标系中的表示

如式 (2.8) 所示，前三个向量是 $w=0$ 的方向向量，表示该坐标系的三个单位向量 $\bar{n}, \bar{o}, \bar{a}$ 的方向，而第四个 $w=1$ 的向量表示该坐标系原点相对于参考坐标系的位置。与单位向量不同，向量 P 的长度十分重要，因而使用比例因子为 1。坐标系也可以由一个没有比例因子的 3×4 矩阵表示，但不常用。

例 2.2 如图 2.7 所示的 F 坐标系位于参考坐标系中 3, 5, 7 的位置，它的 n 轴与 x 轴平行， o 轴相对于 y 轴的角度为 45° ， a 轴相对于 z 轴的角度为 45° 。该坐标系可以表示为：

$$F = \begin{bmatrix} 1 & 0 & 0 & 3 \\ 0 & 0.707 & -0.707 & 5 \\ 0 & 0.707 & 0.707 & 7 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

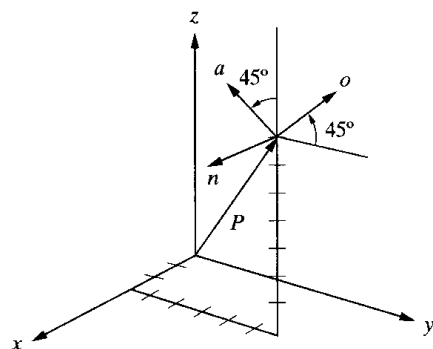


图 2.7 坐标系在空间的表示举例

2.3.5 刚体的表示

一个物体在空间的表示可以这样实现：通过在其上面固连一个坐标系，再将该固连的坐标系在空间表示出来。由于这个坐标系一直固连在该物体上，所以该物体相对于坐标系的位姿是已知的。因此，只要这个坐标系可以在空间表示出来，那么这个物体相对于固定坐标系的位姿也就已知了（如图 2.8 所示）。如前所述，空间坐标系可以用矩阵表示，其中坐标原点以及相对于参考坐标系的表示该坐标系姿态的三个向量也可以由该矩阵表示出来。于是有：

$$F_{object} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

如第 1 章所述，空间中的一个点只有三个自由度，它只能沿三条参考坐标轴移动。但在空间的一个刚体有六个自由度，也就是说，它不仅可以沿着 X, Y, Z 三轴移动，而且还可绕这三个轴转动。因此，要全面地定义空间以物体，需要用 6 条独立的信息来描述物体原点在参考坐标系中相对于三个参考坐标轴的位置，以及物体关于这三个坐标轴的姿态。而式 (2.9)

给出了 12 条信息，其中 9 条为姿态信息，三条为位置信息（排除矩阵中最后一行的比例因子，因为它们没有附加信息）。显然，在该表达式中必定存在一定的约束条件将上述信息数限制为 6。因此，需要用 6 个约束方程将 12 条信息减少到 6 条信息。这些约束条件来自于目前尚未利用的已知的坐标系特性，即：

- 三个向量 $\bar{n}, \bar{o}, \bar{a}$ 相互垂直
- 每个单位向量的长度必须为 1

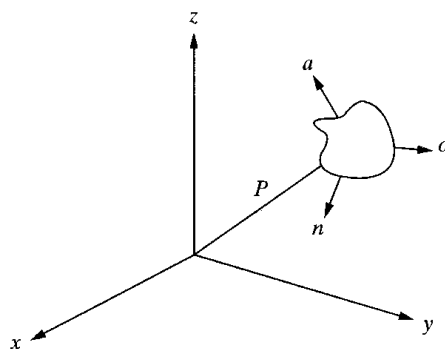


图 2.8 空间物体的表示

我们可以将其转换为以下六个约束方程：

$$(1) \quad \bar{n} \cdot \bar{o} = 0$$

$$(2) \quad \bar{n} \cdot \bar{a} = 0$$

$$(3) \quad \bar{a} \cdot \bar{o} = 0$$

$$(4) \quad |\bar{n}| = 1 \quad (\text{向量的长度必须为 } 1) \quad (2.10)$$

$$(5) \quad |\bar{o}| = 1$$

$$(6) \quad |\bar{a}| = 1$$

因此，只有前述方程成立时，坐标系的值才能用矩阵表示。否则，坐标系将不正确。式 (2.10) 中前三个方程可以换用如下的三个向量的叉积来代替：

$$\bar{n} \times \bar{o} = \bar{a} \quad (2.11)$$

例 2.3 对于下列坐标系，求解所缺元素的值，并用矩阵来表示这个坐标系。

$$F = \begin{bmatrix} ? & 0 & ? & 5 \\ 0.707 & ? & ? & 3 \\ ? & ? & 0 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

解：

显然，表示坐标系原点位置的值 5, 3, 2 对约束方程无影响。注意在三个方向向量中只有三个值是给定的，但这也已足够了。根据式 (2.10)，得：

$$n_x o_x + n_y o_y + n_z o_z = 0 \quad \text{或} \quad n_x(0) + 0.707(o_y) + n_z(o_z) = 0$$

$$n_x a_x + n_y a_y + n_z a_z = 0 \quad \text{或} \quad n_x(a_x) + 0.707(a_y) + n_z(0) = 0$$

$$a_x o_x + a_y o_y + a_z o_z = 0 \quad \text{或} \quad a_x(0) + a_y(o_y) + 0(o_z) = 0$$

$$n_x^2 + n_y^2 + n_z^2 = 1 \quad \text{或} \quad n_x^2 + 0.707^2 + n_z^2 = 1$$

$$o_x^2 + o_y^2 + o_z^2 = 1 \quad \text{或} \quad 0^2 + o_y^2 + o_z^2 = 1$$

$$a_x^2 + a_y^2 + a_z^2 = 1 \quad \text{或} \quad a_x^2 + a_y^2 + 0^2 = 1$$

将这些方程化简得：

$$0.707 o_y + n_z o_z = 0$$

$$n_x a_x + 0.707 a_y = 0$$

$$a_y o_y = 0$$

$$n_x^2 + n_z^2 = 0.5$$

$$o_y^2 + o_z^2 = 1$$

$$a_x^2 + a_y^2 = 1$$

解这六个方程得： $n_x = \pm 0.707, n_z = 0, o_y = 0, o_z = 1, a_x = \pm 0.707$ 和 $a_y = -0.707$ 。应注意， n_x 和 a_x 必须同号。非唯一解的原因是由于给出的参数可能得到两组在相反方向上相互垂直的向量。最终得到的矩阵如下：

$$F = \begin{bmatrix} 0.707 & 0 & 0.707 & 5 \\ 0.707 & 0 & -0.707 & 3 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{或} \quad F = \begin{bmatrix} -0.707 & 0 & -0.707 & 5 \\ 0.707 & 0 & -0.707 & 3 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

由此可见，两个矩阵都满足约束方程的要求。但应注意三个方向向量所表述的值不是任意的，而是受这些约束方程的约束，因此不可任意给矩阵赋值。

同样，可通过 \bar{n} 与 \bar{o} 叉乘并令其等于 \bar{a} ，即 $\bar{n} \times \bar{o} = \bar{a}$ 来求解，表示如下：

$$\begin{vmatrix} i & j & k \\ n_x & n_y & n_z \\ o_x & o_y & o_z \end{vmatrix} = a_x i + a_y j + a_z k$$

或表示为： $i(n_y o_z - n_z o_y) - j(n_x o_z - n_z o_x) + k(n_x o_y - n_y o_x) = a_x i + a_y j + a_z k$

将值代入方程得：

$$i(0.707 o_z - n_z o_y) - j(n_x o_z) + k(n_x o_y) = a_x i + a_y j + 0k$$

同时解下面这三个方程：

$$0.707o_z - n_z o_y = a_x$$

$$-n_x o_z = a_y$$

$$n_x o_y = 0$$

以此来代替三个点乘方程。联立三个单位向量长度的约束方程，便得到六个方程，求解这六个方程，可得到相同的未知参数的解。

2.4 齐次变换矩阵

由于各种原因，变换矩阵应写成方型形式， 3×3 或 4×4 均可。首先，如后面看到的，计算方形矩阵的逆要比计算长方形矩阵的逆容易得多。其次，为使两矩阵相乘，它们的维数必须匹配，即第一矩阵的列数必须与第二矩阵的行数相同。如果两矩阵是方阵则无上述要求。由于要以不同顺序将许多矩阵乘在一起得到机器人运动方程，因此，应采用方阵进行计算。

为保证所表示的矩阵为方阵，如果在同一矩阵中既表示姿态又表示位置，那么可在矩阵中加入比例因子使之成为 4×4 矩阵。如果只表示姿态，则可去掉比例因子得到 3×3 矩阵，或加入第四列全为 0 的位置数据以保持矩阵为方阵。这种形式的矩阵称为齐次矩阵，它们写为：

$$F = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.12)$$

2.5 变换的表示

变换定义为空间的一个运动。当空间的一个坐标系（一个向量、一个物体或一个运动坐标系）相对于固定参考坐标系运动时，这一运动可以用类似于表示坐标系的方式来表示。这是因为变换本身就是坐标系状态的变化（表示坐标系位姿的变化），因此变换可以用坐标系来表示。变换可为如下几种形式中的一种：

- 纯平移
- 绕一个轴的纯旋转
- 平移与旋转的结合

为了解它们的表示方法，我们将逐一进行探讨。

2.5.1 纯平移变换的表示

如果一坐标系（它也可能表示一个物体）在空间以不变的姿态运动，那么该坐标就是纯平移。在这种情况下，它的方向单位向量保持同一方向不变。所有的改变只是坐标系原点相对于参考坐标系的变化，如图 2.9 所示。相对于固定参考坐标系的新的坐标系的位置可以用原来坐标系的原点位置向量加上表示位移的向量求得。若用矩阵形式，新坐标系的表示可以通过坐标系左乘变换矩阵得到。由于在纯平移中方向向量不改变，变换矩阵 T 可以简单地表示为：

$$T = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.13)$$

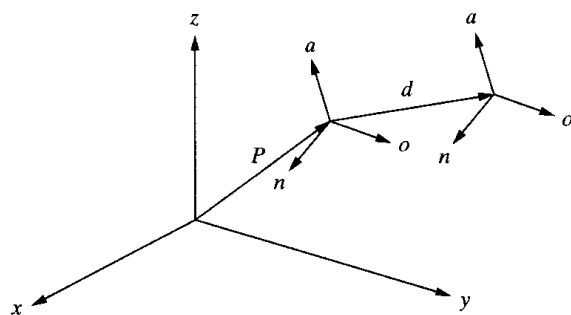


图 2.9 空间纯平移变换的表示

其中 d_x, d_y, d_z 是纯平移向量 \bar{d} 相对于参考坐标系 x, y, z 的三个分量。可以看到，矩阵的前三列表示没有旋转运动（等同于单位阵），而最后一列表示平移运动。新的坐标系位置为：

$$F_{new} = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x + d_x \\ n_y & o_y & a_y & p_y + d_y \\ n_z & o_z & a_z & p_z + d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.14)$$

这个方程也可用符号写为：

$$F_{new} = Trans(d_x, d_y, d_z) \times F_{old} \quad (2.15)$$

首先，如前面所看到的，新坐标系的位置可通过在坐标系矩阵前面左乘变换矩阵得到，后面将看到，无论以何种形式，这种方法对于所有的变换都成立。其次可以注意到，方向向量经过纯平移后保持不变。但是，新的坐标系的位置是 \bar{d} 和 \bar{P} 向量相加的结果。最后应该注意到，齐次变换矩阵与矩阵乘法的关系使得到的新矩阵的维数和变换前相同。

例 2.4 坐标系 F 沿参考坐标系的 x 轴移动 9 个单位，沿 z 轴移动 5 个单位。求新的坐标系位置。

$$F = \begin{bmatrix} 0.527 & -0.574 & 0.628 & 5 \\ 0.369 & 0.819 & 0.439 & 3 \\ -0.766 & 0 & 0.643 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

解：由式（2.14）或式（2.15）得：

$$F_{new} = Trans(d_x, d_y, d_z) \times F_{old} = Trans(9, 0, 5) \times F_{old} \quad \text{和}$$

$$\begin{aligned} F &= \begin{bmatrix} 1 & 0 & 0 & 9 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0.527 & -0.527 & 0.628 & 5 \\ 0.369 & 0.819 & 0.439 & 3 \\ -0.766 & 0 & 0.643 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 0.527 & -0.574 & 0.628 & 14 \\ 0.369 & 0.819 & 0.439 & 3 \\ -0.766 & 0 & 0.643 & 13 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

2.5.2 绕轴纯旋转变换的表示

为简化绕轴旋转的推导，首先假设该坐标系位于参考坐标系的原点并且与之平行，之后将结果推广到其他的旋转以及旋转的组合。

假设坐标系 $(\bar{n}, \bar{o}, \bar{a})$ 位于参考坐标系 $(\bar{x}, \bar{y}, \bar{z})$ 的原点，坐标系 $(\bar{n}, \bar{o}, \bar{a})$ 绕参考坐标系的 x 轴旋转一个角度 θ ，再假设旋转坐标系 $(\bar{n}, \bar{o}, \bar{a})$ 上有一点 P 相对于参考坐标系的坐标为 P_x, P_y, P_z ，相对于运动坐标系的坐标为 P_n, P_o, P_a 。当坐标系绕 x 轴旋转时，坐标系上的点 P 也随坐标系一起旋转。在旋转之前， P 点在两个坐标系中的坐标是相同的（这时两个坐标系位置相同，并且相互平行）。旋转后，该点坐标 P_n, P_o, P_a 在旋转坐标系 (x, y, z) 中保持不变，但在参考坐标系中 P_x, P_y, P_z 却改变了（如图 2.10 所示）。现在要求找到运动坐标系旋转后 P 相对于固定参考坐标系的新坐标。

让我们从 x 轴来观察在二维平面上的同一点的坐标，图 2.10 显示了点 P 在坐标系旋转前后的坐标。点 P 相对于参考坐标系的坐标是 P_x, P_y, P_z ，而相对于旋转坐标系（点 P 所固连的坐标系）的坐标仍为 P_n, P_o, P_a 。

由图 2.11 可以看出， P_x 不随坐标系统 x 轴的转动而改变，而 P_y 和 P_z 却改变了，可以证明：

$$\begin{aligned} P_x &= P_n \\ P_y &= l_1 - l_2 = P_o \cos \theta - P_a \sin \theta \\ P_z &= l_3 + l_4 = P_o \sin \theta + P_a \cos \theta \end{aligned} \quad (2.16)$$

写成矩阵形式为：

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} P_n \\ P_o \\ P_a \end{bmatrix} \quad (2.17)$$

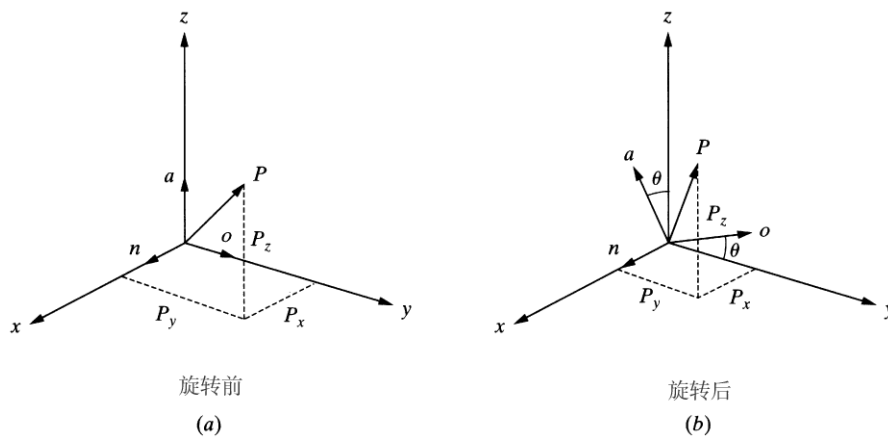


图 2.10 在坐标系旋转前后的点的坐标

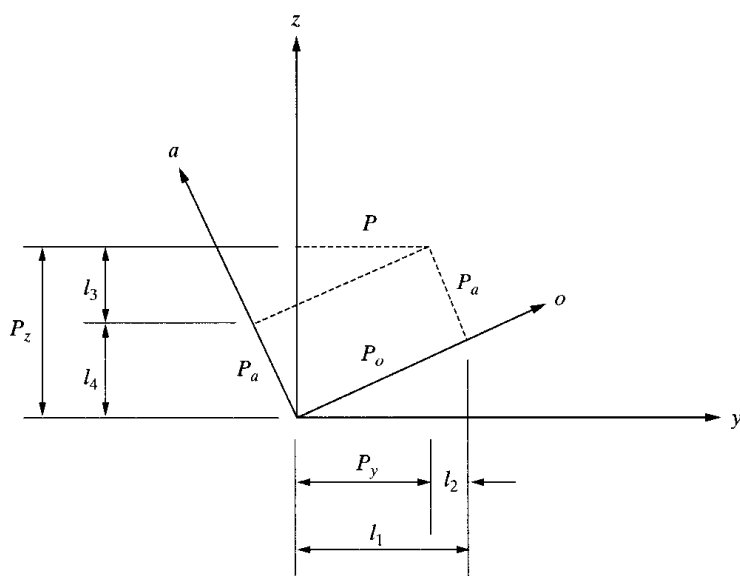


图 2.11 相对于参考坐标系的点的坐标和从 x 轴上观察旋转坐标系

可见，为了得到在参考坐标系中的坐标，旋转坐标系中的点 P （或向量 P ）的坐标必须左乘旋转矩阵。这个旋转矩阵只适用于绕参考坐标系的 x 轴做纯旋转变换的情况，它可表示为：

$$P_{xyz} = Rot(x, \theta) \times P_{noa} \quad (2.18)$$

注意在式（2.17）中，旋转矩阵的第一列表示相对于 x 轴的位置，其值为 1, 0, 0，它表示沿 x 轴的坐标没有改变。

为简化书写，习惯用符号 $C\theta$ 表示 $\cos\theta$ 以及用 $S\theta$ 表示 $\sin\theta$ 。因此，旋转矩阵也可写为：

$$Rot(x, \theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta & -S\theta \\ 0 & S\theta & C\theta \end{bmatrix} \quad (\text{已修正}) \quad (2.19)$$

可用同样的方法来分析坐标系绕参考坐标系 y 轴和 z 轴旋转的情况，可以证明其结果为：

$$Rot(y, \theta) = \begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix} \quad (\text{已修正}) \quad \text{和} \quad Rot(z, \theta) = \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{已修正}) \quad (2.20)$$

式（2.18）也可写为习惯的形式，以便于理解不同坐标系间的关系，为此，可将该变换表示为 ${}^U T_R$ （读作坐标系 R 相对于坐标系 U （Universe）的变换），将 P_{noa} 表示为 ${}^R P$ （ P 相对于坐标系 R ），将 P_{xyz} 表示为 ${}^U P$ （ P 相对于坐标系 U ），式（2.18）可简化为：

$${}^U P = {}^U T_R \times {}^R P \quad (2.21)$$

由上式可见，去掉 R 便得到了 P 相对于坐标系 U 的坐标。

例 2.5 旋转坐标系中有一点 $P(2 \ 3 \ 4)^T$ ，此坐标系绕参考坐标系 x 轴旋转 90 度。求旋转后该点相对于参考坐标系的坐标，并且用图解法检验结果。

解：

由于点 P 固连在旋转坐标系中，因此点 P 相对于旋转坐标系的坐标在旋转前后保持不变。该点相对于参考坐标系的坐标为：

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} P_n \\ P_o \\ P_a \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 2 \\ -4 \\ 3 \end{bmatrix}$$

如图 2.12 所示，根据前面的变换，得到旋转后 P 点相对于参考坐标系的坐标为 2, -4, 3。

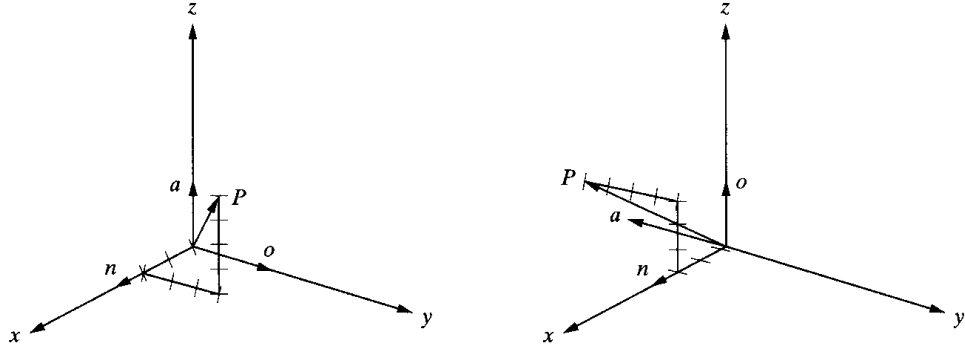


图 2.12 相对于参考坐标系的坐标系旋转

2.5.3 复合变换的表示

复合变换是由固定参考坐标系或当前运动坐标系的一系列沿轴平移和绕轴旋转变换所组成的。任何变换都可以分解为按一定顺序的一组平移和旋转变换。例如，为了完成所要求的变换，可以先绕 x 轴旋转，再沿 x, y, z 轴平移，最后绕 y 轴旋转。在后面将会看到，这个变换顺序很重要，如果颠倒两个依次变换的顺序，结果将会完全不同。

为了探讨如何处理复合变换，假定坐标系 $(\bar{n}, \bar{o}, \bar{a})$ 相对于参考坐标系 (x, y, z) 依次进行了下面三个变换：

- (1) 绕 x 轴旋转 α 度；
- (2) 接着平移 $[l1 \ l2 \ l3]$ （分别相对于 x, y, z 轴）；
- (3) 最后绕 y 轴旋转 β 度。

比如点 P_{noa} 固定在旋转坐标系，开始时旋转坐标系的原点与参考坐标系的原点重合。随着坐标系 $(\bar{n}, \bar{o}, \bar{a})$ 相对于参考坐标系旋转或者平移时，坐标系中的 P 点相对于参考坐标系也跟着改变。如前面所看到的，第一次变换后，P 点相对于参考坐标系的坐标可用下列方程进行计算。

$$P_{1,xyz} = Rot(x, \alpha) \times P_{noa} \quad (2.22)$$

其中， $P_{1,xyz}$ 是第一次变换后该点相对于参考坐标系的坐标。第二次变换后，该点相对于参考坐标系的坐标是：

$$P_{2,xyz} = Trans(l1, l2, l3) \times P_{1,xyz} = Trans(l1, l2, l3) \times Rot(x, \alpha) \times P_{noa}$$

同样，第三次变换后，该点相对于参考坐标系的坐标为：

$$P_{xyz} = P_{3,xyz} = Rot(y, \beta) \times P_{2,xyz} = Rot(y, \beta) \times Trans(l1, l2, l3) \times Rot(x, \alpha) \times P_{noa}$$

可见，每次变换后该点相对于参考坐标系的坐标都是通过用每个变换矩阵左乘该点的坐

标得到的。当然，矩阵的顺序不能改变。同时还应注意，对于相对于参考坐标系的每次变换，矩阵都是左乘的。因此，矩阵书写的顺序和进行变换的顺序正好相反。

例 2.6 固连在坐标系 $(\bar{n}, \bar{o}, \bar{a})$ 上的点 $P(7 \ 3 \ 2)^T$ 经历如下变换，求出变换后该点相对于参考坐标系的坐标。

- (1) 绕 z 轴旋转 90 度；
- (2) 接着绕 y 轴旋转 90 度；
- (3) 接着再平移 $[4, -3, 7]$ 。

解：

表示该变换的矩阵方程为：

$$P_{xyz} = \text{Trans}(4, -3, 7) \text{Rot}(y, 90) P_{noa} =$$

$$\begin{bmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 7 \\ 3 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 4 \\ 10 \\ 1 \end{bmatrix}$$

从图 2.13 可以看到， $(\bar{n}, \bar{o}, \bar{a})$ 坐标系首先绕 z 轴旋转 90 度，接着绕 y 轴旋转，最后相对于参考坐标系的 x, y, z 轴平移。坐标系中的 P 点相对于 $\bar{n}, \bar{o}, \bar{a}$ 轴的位置如图所示，最后该点在 x, y, z 轴上的坐标分别为 $4+2=6$ ， $-3+7=4$ ， $7+3=10$ 。请确认也能从图中理解上述结果。

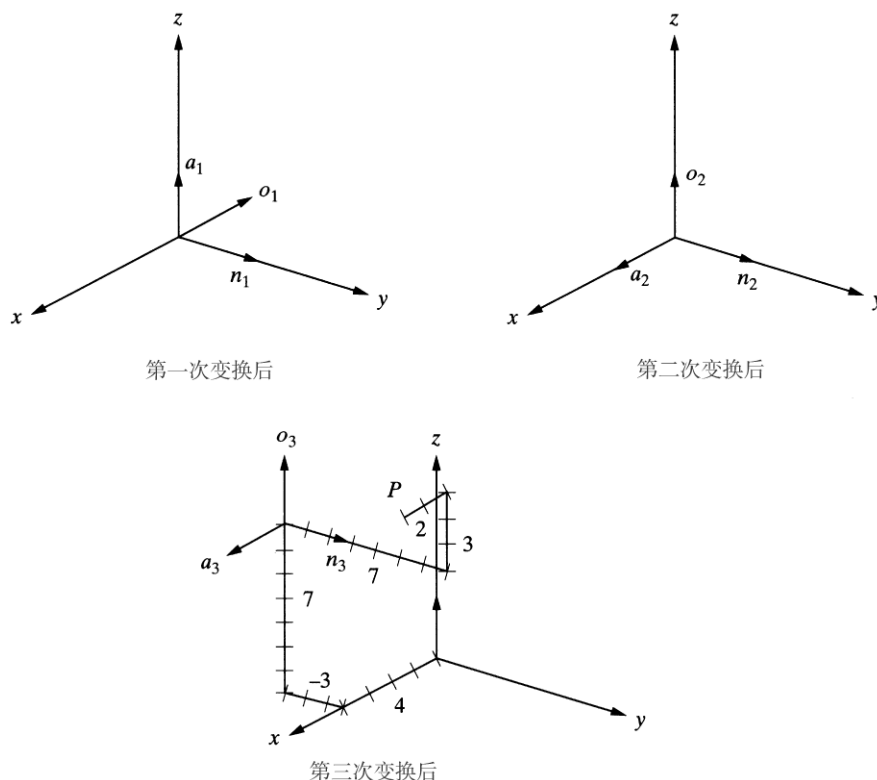


图 2.13 三次顺序变换的结果

例 2.7 根据上例，假定 $(\bar{n}, \bar{o}, \bar{a})$ 坐标系上的点 $P(7 \ 3 \ 2)^T$ 经历相同变换，但变换按如下不同顺序进行，求出变换后该点相对于参考坐标系的坐标。

- (1) 绕 z 轴旋转 90 度；

- (2) 接着平移[4, -3, 7];
 (3) 接着再绕 y 轴旋转 90 度。

解:

表示该变换的矩阵方程为:

$$P_{xyz} = Rot(y, 90) Trans(4, -3, 7) Rot(z, 90) P_{noa} =$$

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 7 \\ 3 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 9 \\ 4 \\ -1 \\ 1 \end{bmatrix}$$

不难发现, 尽管所有的变换与例 2.6 完全相同, 但由于变换的顺序变了, 该点最终坐标与前例完全不同。用图 2.14 可以清楚地说明这点, 这时可以看出, 尽管第一次变换后坐标系的变化与前例完全相同, 但第二次变换后结果就完全不同, 这是由于相对于参考坐标系轴的平移使得旋转坐标系 $(\bar{n}, \bar{o}, \bar{a})$ 向外移动了。经第三次变换, 该坐标系将绕参考坐标系 y 轴旋转, 因此向下旋转了, 坐标系上点 P 的位置也显示在图中。

请证明该点相对于参考坐标系的坐标为 $7+2=9$, $-3+7=4$ 和 $-4+3=-1$, 它与解析的结果相同。

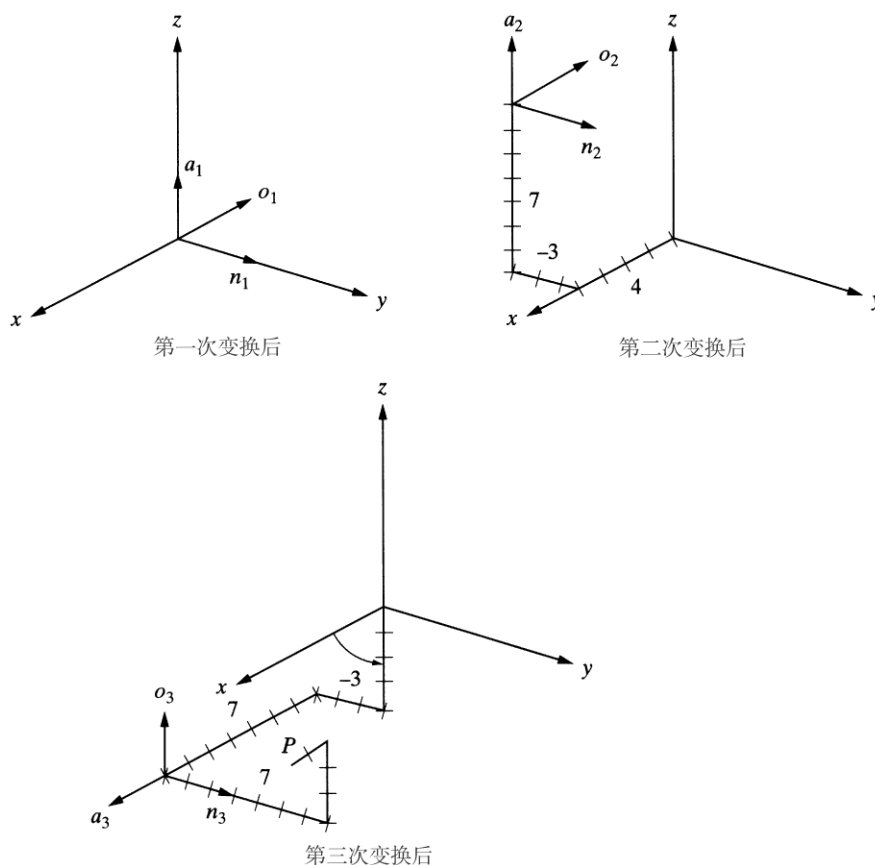


图 2.14 变换的顺序改变将改变最终结果

2.5.4 相对于旋转坐标系的变换

到目前为止, 本书所讨论的所有变换都是相对于固定参考坐标系。也就是说, 所有平移、旋转和距离 (除了相对于运动坐标系的点的位置) 都是相对参考坐标系轴来测量的。然而事实上, 也有可能做相对于运动坐标系或当前坐标系的轴的变换。例如, 可以相对于运动坐标系 (也就是当前坐标系) 的 \bar{n} 轴而不是参考坐标系的 x 轴旋转 90 度。为计算当前坐标系中的点的坐标相对于参考坐标系的变化, 这时需要右乘变换矩阵而不是左乘。由于运动坐标系中的点或物体的位置总是相对于运动坐标系测量的, 所以总是右乘描述该点或物体的位置矩阵。

例 2.8 假设与例 2.7 中相同的点现在进行相同的变换，但所有变换都是相对于当前的运动坐标系，具体变换出如下。求出变换完成后该点相对于参考坐标系的坐标。

- (1) 绕 \bar{a} 轴旋转 90 度；
- (2) 然后沿 $\bar{n}, \bar{o}, \bar{a}$ 轴平移[4, -3, 7]；
- (3) 接着绕 \bar{o} 轴旋转 90 度。

解：

在本例中，因为所作变换是相对于当前坐标系的，因此右乘每个变换矩阵，可得表示该坐标的方程为：

$$P_{xyz} = Rot(\bar{a}, 90) Trans(4, -3, 7) Rot(\bar{o}, 90) P_{noa} =$$

$$\begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 7 \\ 3 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 6 \\ 0 \\ 1 \end{bmatrix}$$

如所期望的，结果与其他各例完全不同，不仅因为所作变换是相对于当前坐标系的，而且也因为矩阵顺序的改变。下面的图展示了这一结果，应注意它是怎样相对于当前坐标来完成这个变换的。

同时应注意，在当前坐标系中 p 点的坐标 7, 3, 2 是变换后得到相对于参考坐标系的坐标 0, 6, 0 的。

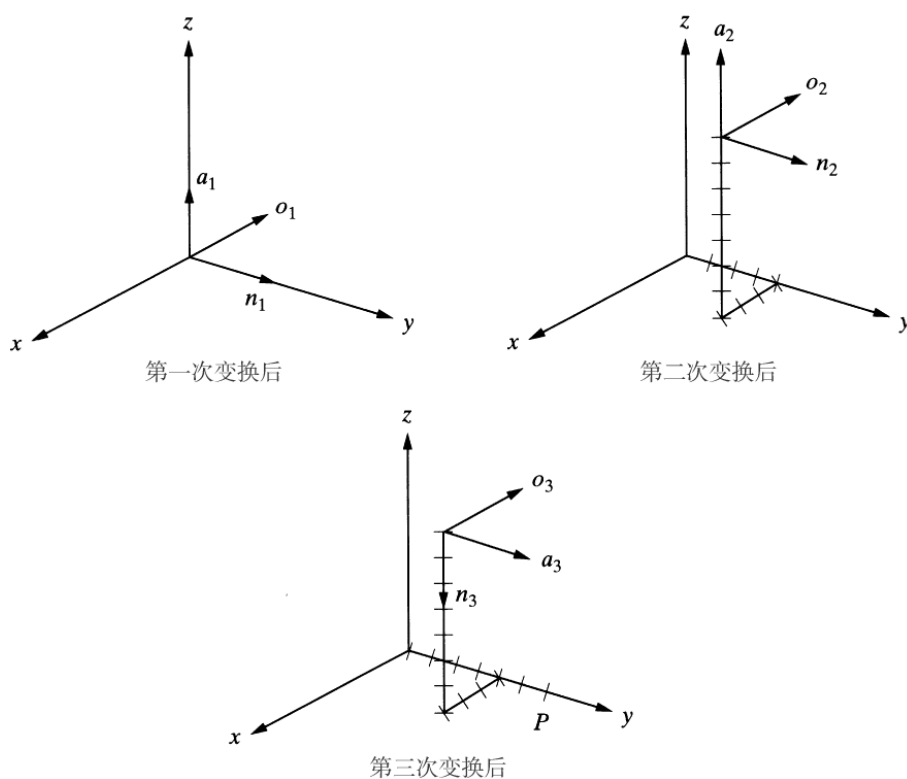


图 2.15 相对于当前坐标系的变换

例 2.9 坐标系 B 绕 x 轴旋转 90°，然后沿当前坐标系 a 轴做了 3 英寸的平移，然后在绕 z 轴旋转 90，最后沿当前坐标系 o 轴做 5 英寸的平移。

(a) 写出描述该运动的方程。

(b)求坐标系中的点 p (1, 5, 4) 相对于参考坐标系的最终位置。

解:

在本例中, 相对于参考坐标系以及当前坐标系的运动是交替进行的。

(a)相应地左乘或右乘每个运动矩阵, 得到:

$${}^uT_B = \text{Rot}(z, 90) \text{Rot}(x, 90) \text{Trans}(0, 0, 3) \text{Trans}(0, 5, 0)$$

(b)带入具体的矩阵并将它们相乘, 得到:

$${}^uP = {}^uT_B \times {}^BP = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 5 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 1 & 7 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 10 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.6 变换矩阵的逆

正如前面所提到的, 在机器人分通过析中有很多地方要用到矩阵的逆, 在下面的例子中可以看到一种涉及变换矩阵的情况。在图 2.16 中, 假设机器人要在零件 p 上钻孔而须向零件 p 处移动。机器人基座相对于参考坐标系 u 的位置用坐标系 R 来描述, 机器人手用坐标系 H 来描述, 末端执行器 (即用来钻孔的钻头的末端) 用坐标系 E 来描述, 零件的位置用坐标系 P 来描述。钻孔的点的位置于参考坐标系 U 可以通过两个独立的路径发生联系: 一个是通过该零件的路径, 另一个是通过机器人的路径。因此, 可以写出下面的方程:

$${}^UT_E = {}^UT_R {}^RT_H {}^HT_E = {}^UT_P {}^PT_E \quad (2.24)$$

这就是说, 该零件中点 E 的位置可以通过从 U 变换到 P, 并从 P 变换到 E 来完成, 或者从 U 变换到 R, 从 R 变换到 H, 再从 H 变换到 E。

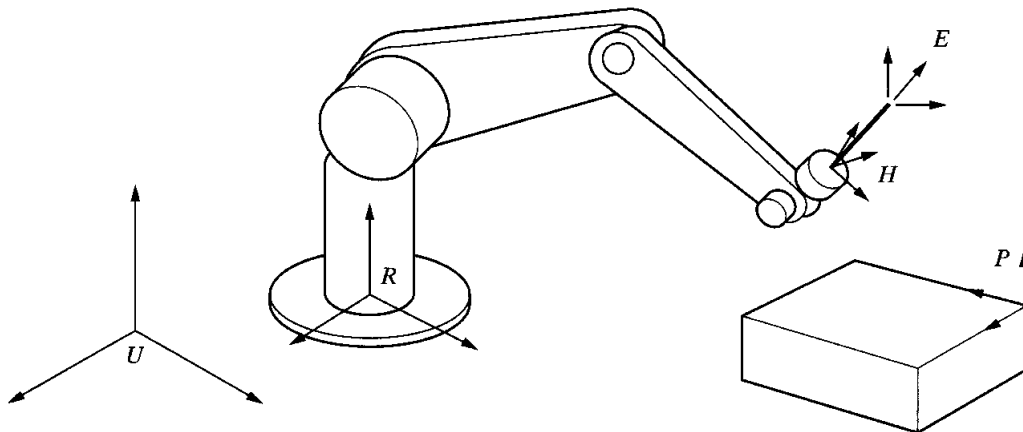


图 2.16 全局坐标系、机器人坐标系、手坐标系、零件坐标系及末端执行器坐标系

事实上, 由于在任何情况下机器人的基座位置在安装时就是已知的, 因此变换 UT_R (坐标系 R 相对于坐标系 U 的变换) 是已知的。比如, 一个机器人安装在一个工作台上, 由于它被紧固在工作台上, 所以它的基座的位置是已知的。即使机器人是可移的或放在传送带上, 因为控制

器始终掌控着机器人基座的运动，因此它在任意时刻的位置也是已知的。由于用于末端执行器的任何器械都是已知的，而且其尺寸和结构也是已知的，所以 ${}^H T_E$ （机器人末端执行器相对于机器人手的变换）也是已知的。此外， ${}^U T_P$ （零件相对于全局坐标系的变换）也是已知的，还必须要知道将在其上面钻孔的零件的位置，该位置可以通过将该零件放在钻模上，让后用照相机，视觉系统，传送带，传感器或其他类似仪器来确定。最后需要知道零件上钻孔的位置，所以 ${}^P T_E$ 也是已知的。此时，唯一未知的变换就是 ${}^R T_H$ （机器人手相对于机器人基座的变换）。因此，必须找出机器人的关节变量（机器人旋转关节的角度以及滑动关节的连杆长度），以便将末端执行器定位在要钻孔的位置上。可见，必须要计算出这个变换，它指出机器人需要完成的工作。后面将用所求出的变换来求解机器人关节的角度和连杆的长度。

不能像在代数方程中那样来计算这个矩阵，即不能简单的用方程的右边除以方程的左边，而应该用合适的矩阵的逆并通过左乘或右乘来将它们从左边去调。因此有：

$$({}^U T_R)^{-1}({}^U T_R {}^R T_H {}^H T_E)({}^H T_E)^{-1} = ({}^U T_R)^{-1}({}^U T_P {}^P T_E)({}^H T_E)^{-1} \quad (2.25)$$

由于 $({}^U T_R)^{-1}({}^U T_R) = 1$ 和 $({}^H T_E)({}^H T_E)^{-1} = 1$ ，式（2.25）的左边可简化为 ${}^R T_H$ ，于是得：

$${}^R T_H = {}^U T_R^{-1} {}^U T_P {}^P T_E {}^H T_E^{-1} \quad (2.26)$$

该方程的正确性可以通过认为 ${}^E T_H$ 与 $({}^H T_E)^{-1}$ 相同来加以检验。因此，该方程可重写为：

$${}^R T_H = {}^U T_R^{-1} {}^U T_P {}^P T_E {}^H T_E^{-1} = {}^R T_U {}^U T_P {}^P T_E {}^E T_H = {}^R T_H \quad (2.27)$$

显然为了对机器人运动学进行分析，需要能够计算变换矩阵的逆。

我们来看看关于 x 轴的简单旋转矩阵的求逆计算情况。关于 x 轴的旋转矩阵是：

$$Rot(x, \theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta & -S\theta \\ 0 & S\theta & C\theta \end{bmatrix} \quad (2.28)$$

必须采用一下的步骤来计算矩阵的逆：

- 计算矩阵的行列式；
- 将矩阵转置；
- 将转置矩阵的每个元素用它的子行列式（伴随矩阵）代替；
- 用转换后的矩阵除以行列式。

将上面的步骤用到该旋转，得到：

$$\Delta = 1(C^2\theta + S^2\theta) + 0 = 1$$

$$Rot(x, \theta)^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta & S\theta \\ 0 & -S\theta & C\theta \end{bmatrix}$$

现在计算每一个子行列式（伴随矩阵）。例如，元素 2, 2 的子行列式是 $C\theta - 0 = C\theta$ ，元素 1, 1 的子行列式 $C^2\theta + S^2\theta = 1$ 。可以注意到，这里的每一个元素的子行列式与其本身相同，因此有：

$$Rot(x, \theta)_{minor}^T = Rot(x, \theta)^T$$

由于原旋转矩阵的行列式为 1，因此用 $Rot(x, \theta)_{minor}^T$ 矩阵除以行列式仍得出相同的结果。

因此，关于 x 轴的旋转矩阵的逆的行列式与它的转置矩阵相同，即：

$$Rot(x, \theta)^{-1} = Rot(x, \theta)^T \quad (2.29)$$

当然，如果采用附录 A 中提到的第二种方法也能得到同样的结果。具有这种特征的矩阵称为酉矩阵，也就是说所有的旋转矩阵都是酉矩阵。因此，计算旋转矩阵的逆就是将该矩阵转置。可以证明，关于 y 轴和 z 轴的旋转矩阵同样也是酉矩阵。

应注意，只有旋转矩阵才是酉矩阵。如果一个矩阵不是一个简单的旋转矩阵，那么它也许就不是酉矩阵。

以上结论只对简单的不表示位置的 3×3 旋转矩阵成立。对一个齐次的 4×4 变换矩阵而言，它的求逆可以将矩阵分为两部分。矩阵的旋转部分仍是酉矩阵，只需简单的转置；矩阵的位置部分是向量 \bar{P} 分别与 $\bar{n}, \bar{o}, \bar{a}$ 向量点积的负值，其结果为：

$$F = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.30)$$

$$F = \begin{bmatrix} n_x & n_y & n_z & -\bar{P} \cdot \bar{n} \\ o_x & o_y & o_z & -\bar{P} \cdot \bar{o} \\ a_x & a_y & a_z & -\bar{P} \cdot \bar{a} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

如上所示，矩阵的旋转部分是简单的转置，位置的部分由点乘的负值代替，而最后一行（比例因子）则不受影响。这样做对于计算变换矩阵的逆是很有帮助的，而直接计算 4×4 矩阵的逆是一个很冗长的过程。

例 2.10 计算表示 $Rot(x, 40^\circ)^{-1}$ 的矩阵

解：绕 x 轴旋转 40° 的矩阵为：

$$Rot(x, 40^\circ) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.766 & -0.643 & 0 \\ 0 & 0.643 & 0.766 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

其矩阵的逆是：

$$Rot(x, 40^\circ)^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.766 & 0.643 & 0 \\ 0 & -0.643 & 0.766 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

需注意的是，由于矩阵的位置向量为 0，它与 $\bar{n}, \bar{o}, \bar{a}$ 向量的点积也为零。

例 2.11 计算如下变换矩阵的逆。

$$T = \begin{bmatrix} 0.5 & 0 & 0.866 & 3 \\ 0.866 & 0 & -5 & 2 \\ 0 & 1 & 0 & 5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

解：

根据先前的计算，变换矩阵的逆是：

$$T^{-1} = \begin{bmatrix} 0.5 & 0.866 & 0 & -(3 \times 0.5 + 2 \times 0.866 + 5 \times 0) \\ 0 & 0 & 1 & -(3 \times 0 + 2 \times 0 + 5 \times 1) \\ 0.866 & -0.5 & 0 & -(3 \times 0.866 + 2 \times -0.5 + 5 \times 0) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0.5 & 0.866 & 0 & -3.23 \\ 0 & 0 & 1 & -5 \\ 0.866 & -0.5 & 0 & -1.598 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

可以证明 TT^{-1} 是单位阵。

例 2.12 一个具有六个自由度的机器人的第五个连杆上装有照相机，照相机观察物体并测定它相对于照相机坐标系的位置，然后根据以下数据来确定末端执行器要到达物体所必须完成的运动。

$${}^5T_{cam} = \begin{bmatrix} 0 & 0 & -1 & 3 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^5T_H = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{cam}T_{obj} = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^HT_E = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

解：

参照式 (2.24)，可以写出一个与它类似的方程，它将不同的变换和坐标系联系在一起。

$${}^RT_5 \times {}^5T_H \times {}^HT_E \times {}^ET_{obj} = {}^RT_5 \times {}^5T_{cam} \times {}^{cam}T_{obj}$$

由于方程两边都有 RT_5 ，所以可以将它消去。除了 ${}^ET_{obj}$ 之外所有其他矩阵都是已知的，所以：

$${}^ET_{obj} = {}^HT_E^{-1} \times {}^5T_H^{-1} \times {}^5T_{cam} \times {}^{cam}T_{obj} = {}^ET_H \times {}^HT_5 \times {}^5T_{cam} \times {}^{cam}T_{obj}$$

$${}^H T_E^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^5 T_H^{-1} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

将矩阵及矩阵的逆代入前面的方程，得：

$${}^E T_{obj} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & -1 & 3 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^E T_{obj} = \begin{bmatrix} -1 & 0 & 0 & -2 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & -1 & -4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.7 机器人的正逆运动学

假设有一个构型已知的机器人，即它的所有连杆长度和关节角度都是已知的，那么计算机器人手的位姿就称为正运动学分析。换言之，如果已知所有机器人关节变量，用正运动学方程就能计算任一瞬间机器人的位姿。然而，如果想要将机器人的手放在一个期望的位姿，就必须知道机器人的每一个连杆的长度和关节的角度，才能将手定位在所期望的位姿，这就叫做逆运动学分析，也就是说，这里不是把已知的机器人变量代入正向运动学方程中，而是要设法找到这些方程的逆，从而求得所需的关节变量，使机器人放置在期望的位姿。事实上，逆运动学方程更为重要，机器人的控制器将用这些方程来计算关节值，并以此来运行机器人到达期望的位姿。下面首先推导机器人的正运动学方程，然后利用这些方程来计算逆运动学方程。

对正运动学，必须推导出一组与机器人特定构型（将构件组合在一起构成机器人的方法）有关的方程，以使得将有关的关节和连杆变量代入这些方程就能计算出机器人的位姿，然后可用这些方程推出逆运动学方程。

根据第一章中的相关内容，要确定一个刚体在空间的位姿，须在物体上固连一个坐标系，然后描述该坐标系的原点位置和它三个轴的姿态，总共需要六个自由度或六条信息来完整地定义该物体的位姿。同理，如果要确定或找到机器人手在空间的位姿，也必须在机器人手上固连一个坐标系并确定机器人手坐标系的位姿，这正是机器人正运动学方程所要完成的任务。换言之，根据机器人连杆和关节的构型配置，可用一组特定的方程来建立机器人手的坐标系和参考坐标系之间的联系。图 2.17 所示为机器人手的坐标系、参考坐标系以及它们的相对位姿，两个坐标系之间的关系与机器人的构型有关。当然，机器人可能有许多不同的构型，后面将会看到将如何根据机器人的构型来推导出与这两个坐标系相关的方程。

为使过程简化，可分别分析位置和姿态问题，首先推导出位置方程，然后再推导出姿态方程，再将两者结合在一起而形成一组完整的方程。最后，将看到关于 Denavit-Hartenberg 表示法的应用，该方法可用于对任何机器人构型建模。

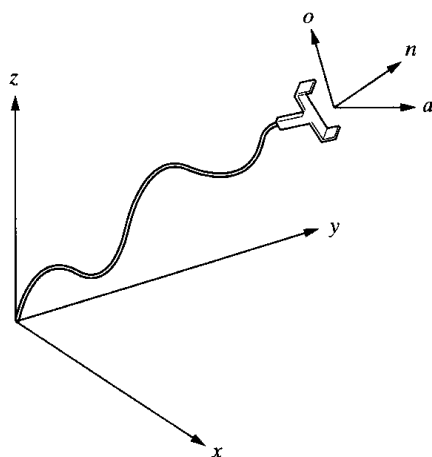


图 2.17 机器人的手坐标系相对于参考坐标系

2.7.1 位置的正逆运动学方程

对于机器人的定位，可以通过相对于任何惯用坐标系的运动来实现。比如，基于直角坐标系对空间的一个点定位，这意味着有三个关于 x, y, z 轴的线性运动，此外，如果用球坐标来实现，就意味着需要有一个线性运动和两个旋转运动。常见的情况有：

- (a)笛卡尔（台架，直角）坐标
- (b)圆柱坐标
- (c)球坐标
- (d)链式(拟人或全旋转)坐标

2.7.1.1 笛卡尔(台架，直角)坐标

这种情况下有三个沿 x, y, z 轴的线性运动，这一类型的机器人所有的驱动机构都是线性的（比如液压活塞或线性动力丝杠），这时机器人手的定位是通过三个线性关节分别沿三个轴的运动来完成的（如图 2.18 所示）。台架式机器人基本上就是一个直角坐标机器人，只不过是机器人固连在一个朝下的直角架上。IBM7565 机器人就是一个台架式直角坐标机器人。

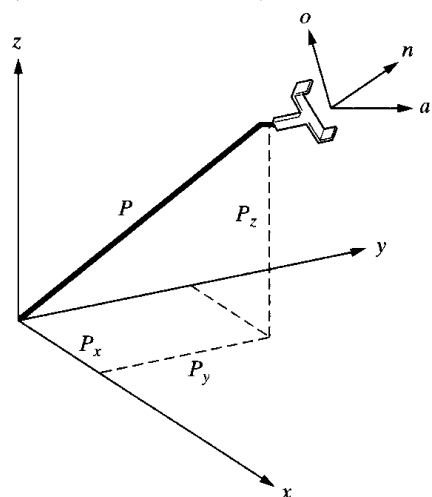


图 2.18 直角坐标

当然，如果没有旋转运动，表示向 P 点运动的变换矩阵是一种简单的平移变换矩阵，下面将可以看到这一点。注意这里只涉及坐标系原点的定位，而不涉及姿态。在直角坐标系中，表示机器人手位置的正运动学变换矩阵为：

$${}^R T_P = T_{cart} = \begin{bmatrix} 1 & 0 & 0 & P_x \\ 0 & 1 & 0 & P_y \\ 0 & 0 & 1 & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.31)$$

其中 ${}^R T_P$ 是参考坐标系与手坐标系原点 P 的变换矩阵，而 T_{cart} 表示直角坐标变换矩阵。对于逆运动学的求解，只需简单地设定期望的位置等于 P 。

例 2.13 要求笛卡尔坐标机器人手坐标系原点定位在点 $P=[3 \ 4 \ 7]^T$ ，计算所需要的笛卡儿坐标运动。

解：

设定正运动学方程用式 (2.31) 中的 ${}^R T_P$ 矩阵表示，根据期望的位置可得知如下结果：

$${}^R T_P = \begin{bmatrix} 1 & 0 & 0 & P_x \\ 0 & 1 & 0 & P_y \\ 0 & 0 & 1 & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 3 \\ 0 & 1 & 0 & 4 \\ 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ 或 } P_x = 3, P_y = 4, P_z = 7$$

2.7.1.2 圆柱坐标

圆柱型坐标系包括两个线性平移运动和一个旋转运动。其顺序为：先沿 x 轴移动 r ，再绕 z 轴旋转 α 角，最后沿 z 轴移动 l ，如图 2.19 所示。这三个变换建立了手坐标系与参考坐标系之间的联系。由于这些变换都是相对于全局参考坐标系的坐标轴的，因此由这三个变换所产生的总变换可以通过依次左乘每一个矩阵而求得：

$${}^R T_P = T_{cyl}(r, \alpha, l) = Trans(0, 0, l) Rot(z, \alpha) Trans(r, 0, 0) \quad (2.32)$$

$${}^R T_P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} C\alpha & -S\alpha & 0 & 0 \\ S\alpha & C\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & r \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.33)$$

$${}^R T_P = T_{cyl} = \begin{bmatrix} C\alpha & -S\alpha & 0 & rC\alpha \\ S\alpha & C\alpha & 0 & rS\alpha \\ 0 & 0 & 1 & l \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

经过一系列变换后，前三列表示了坐标系的姿态，然而我们只对坐标系的原点位置即最后一列感兴趣。显然，在圆柱型坐标运动中，由于绕 z 轴旋转了 α 角，运动坐标系的姿态也将改变，这一改变将在后面讨论。

实际上，可以通过绕 $\bar{n}, \bar{o}, \bar{a}$ 坐标系中的 \bar{a} 轴旋转 $-\alpha$ 角度使坐标系回转到和初始参考坐标系平行的状态，它等效于圆柱坐标矩阵右乘旋转矩阵 $(\bar{a}, -\alpha)$ ，其结果是，该坐标系的位置仍在同一地方，但其姿态再次平行于参考坐标系，如下所示：

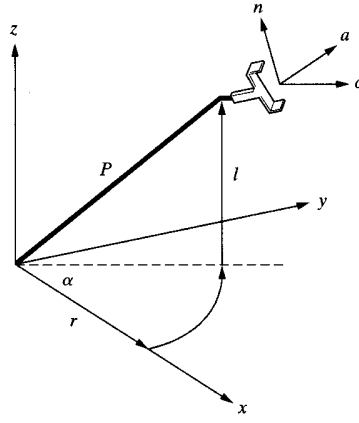


图 2.19 圆柱坐标

$$T_{cyl} \times Rot(z, -\alpha) = \begin{bmatrix} C\alpha & -S\alpha & 0 & rC\alpha \\ S\alpha & C\alpha & 0 & rS\alpha \\ 0 & 0 & 1 & l \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} C(-\alpha) & -S(-\alpha) & 0 & 0 \\ S(-\alpha) & C(-\alpha) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & rC\alpha \\ 0 & 1 & 0 & rS\alpha \\ 0 & 0 & 1 & l \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

由此可见，运动坐标系的原点位置没有改变，但它转回到了与参考坐标系平行的状态。需要注意的是，最后的旋转是绕本地坐标系的 a 轴的，其目的是为了不发生坐标系位置的任何改变，而只改变姿态。

例 2.14 假设要将圆柱坐标机器人手坐标系的原点放在 $[3 \ 4 \ 7]^T$ ，计算该机器人的关节变量。

解：

根据式 (2.33) 的 T_{cyl} 矩阵，将手坐标系原点的位置分量设置为期望值，可以得到：

$$\begin{aligned} l &= 7 \\ rC\alpha &= 3 \\ rS\alpha &= 4 \end{aligned}$$

于是有： $\tan \alpha = \frac{4}{3}$ 和 $\alpha = 53.1^\circ$

将 α 代入其中任何一个方程，可得 $r = 5$ ，最终结果是：

$$r = 5, \alpha = 53.1^\circ, l = 7$$

应注意：必须确保在机器人运动学中计算的角度位于正确的象限。在这个例子中，请注意 $rC\alpha$ 和 $rS\alpha$ 都是正的，并且 r 也是正的，这样角度 α 便在第一象限，且为 53.1° 。

2.7.1.3 球坐标

如图 2.20 所示，球坐标系由一个线性运动和两个旋转运动组成，运动顺序为：先沿 z 轴平移 r ，再绕 y 轴旋转 β 并绕 z 轴旋转 γ 。这三个变换建立了手坐标系与参考坐标系之间的联系。由于这些变换都是相对于全局参考坐标系的坐标轴的，因此有这三个变换所产生的总变换可以通过一次左乘每一个矩阵而求得：

$${}^R T_P = T_{sph}(r, \beta, \gamma) = Rot(z, \gamma) Rot(y, \beta) Trans(0, 0, \gamma) \quad (2.34)$$

$${}^R T_P = \begin{bmatrix} C\gamma & -S\gamma & 0 & 0 \\ S\gamma & C\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} C\beta & 0 & S\beta & 0 \\ 0 & 1 & 0 & 0 \\ -S\beta & 0 & C\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & r \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^R T_P = T_{sph} = \begin{bmatrix} C\beta \cdot C\gamma & -S\gamma & S\beta \cdot C\gamma & rS\beta \cdot C\gamma \\ C\beta \cdot S\gamma & C\gamma & S\beta \cdot S\gamma & rS\beta \cdot S\gamma \\ -S\beta & 0 & C\beta & rC\beta \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.35)$$

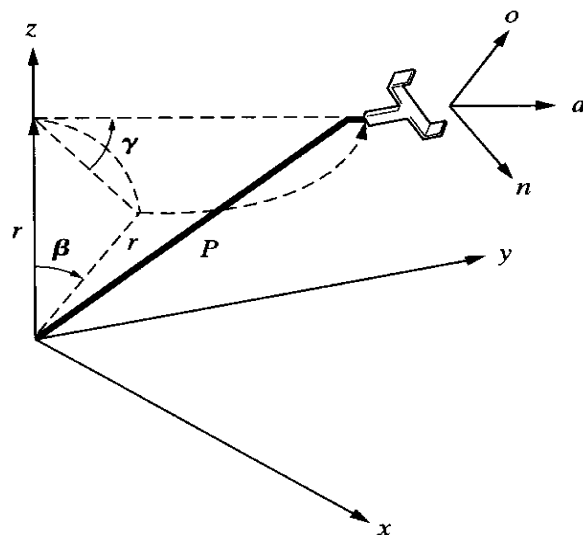


图 2.20 球坐标

前三列表是了经过一系列变换后的坐标系的姿态，而最后一列则表示了坐标系原点的位置。以后还要进一步讨论该矩阵的姿态部分。

这里也可回转最后一个坐标系，使它与参考坐标系平行。这一问题将作为练习留给读者，要求找出正确的运动顺序来获得正确的答案。

球坐标的逆运动学方程比简单的直角坐标和圆柱坐标更复杂，因为两个角度 β 和 γ 是耦合的。让我们通过下例来说明如何求解球坐标的逆运动学方程。

例2.15 假设要将球坐标机器人手坐标系原点放在 $[3, 4, 7]^T$ ，计算机器人的关节变量。

解：

根据式 (2.35) 的 T_{sph} 矩阵，将手坐标系原点的位置分量设置为期望值，可以得到：

$$rS\beta C\gamma = 3$$

$$rS\beta S\gamma = 4$$

$$rC\beta = 7$$

由第三个方程，我们得出 $C\beta$ 是正数，但没有关于 $S\beta$ 是正或负的信息。将前两个方程彼此相除，因为不知道 $S\beta$ 的实际符号是什么，因此可能会有两个解。下面的方法给出了两个可

能的解，后面还必须对这最后的结果进行检验以确保它们是正确的。

$$\begin{aligned}\tan \gamma &= \frac{4}{3} \rightarrow & \gamma &= 53.1^\circ \quad \text{或} \quad 233.1^\circ \\ & & S\gamma &= 0.8 \quad \text{或} \quad -0.8 \\ & & C\gamma &= 0.6 \quad \text{或} \quad -0.6 \\ \gamma S\beta &= \frac{3}{0.6} = 5 & & \text{或} \quad -5 \\ \gamma C\beta &= 7 & \beta &= 35.5^\circ \quad \text{或} \quad -35.5^\circ \\ \gamma &= 8.6\end{aligned}$$

可以对这两组解进行检验并正是这两组解都能满足所有的位置方程。如果沿给定的三维坐标轴旋转这些角度，物理上的确能到达同一点。然而必须注意，其中只有一组解能满足姿态方程。换句话说，前两种解将产生同样的位置，但处于不同的姿态。由于目前并不关心手坐标系在这点的姿态，因此两个位置解都是正确的。实际上，由于不能对三自由度的机器人指定姿态，所以无法确定两个解中哪一个和特定的姿态有关。

2.7.1.4 链式坐标

如图 2.21 所示，链式坐标由三个旋转组成。后面在讨论 Denavit-Hartenberg 表示法时，将推导链式坐标的矩阵表示法。

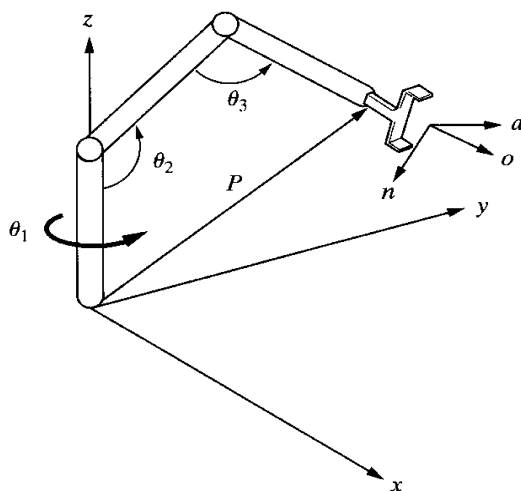


图 2.21 链式坐标

2.7.2 姿态的正逆运动学方程

假设固连在机器人手上的运动坐标系已经运动到期望的位置上，但它仍然平行于参考坐标系，或者假设其姿态并不是所期望的，下一步是要在不改变位置的情况下，适当地旋转坐标系而使其达到所期望的姿态。合适的旋转顺序取决于机器人手腕的设计以及关节装配在一起的方式。考虑以下三中常见的构型配置：

- (a) 滚动角、俯仰角、偏航（RPY）角
- (b) 欧拉角
- (c) 链式关节

2.7.2.1 滚动角、俯仰角和偏航角（绕固定坐标系，顺序由右向左，具体见末尾）

这是分别绕当前 $\bar{a}, \bar{o}, \bar{n}$ 轴的三个旋转顺序，能够把机器人的手调整到所期望的姿态。此时，假定当前的坐标系平行于参考坐标系，于是机器人手的姿态在 RPY（滚动角、俯仰角、偏航

角)运动前与参考坐标系相同。如果当前坐标系不平行于参考坐标系,那么机器人手最终的姿态将会是先前的姿态与 RPY 右乘的结果。

因为不希望运动坐标系原点的位置有任何改变(它已被放在一个期望的位置上,所以只需要旋转她到所期望的姿态),所以 RPY 的旋转运动都是相对于当前的运动轴的。否则。如前面所看到的,运动坐标系的位置将会改变。于是,右乘所有由 RPY 和其他旋转所产生的与姿态改变有关的矩阵。

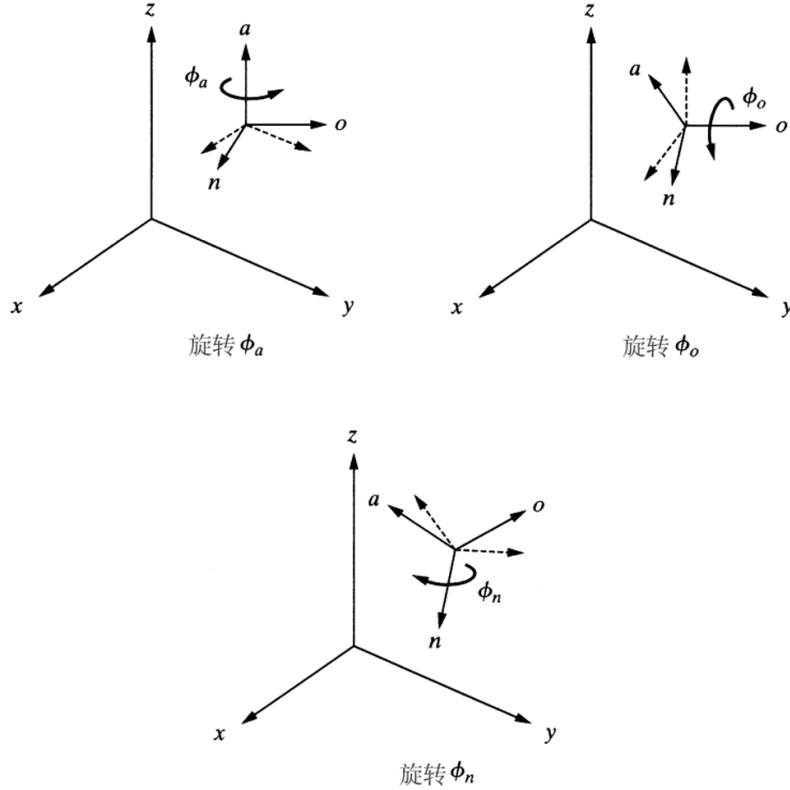


图 2.22 绕固定坐标轴的 RPY 旋转(图有错误)

参考图 2.22, 可看到 RPY 旋转包括以下几种:

- 绕 \bar{a} 轴(运动坐标系的 z 轴)旋转 ϕ_a 叫滚动;
- 绕 \bar{o} 轴(运动坐标系的 y 轴)旋转 ϕ_o 叫俯仰;
- 绕 \bar{n} 轴(运动坐标系的 x 轴)旋转 ϕ_n 叫偏航。

表示 RPY 姿态变化的矩阵为:

$$\begin{aligned}
 // \text{RPY}(\phi_a, \phi_o, \phi_n) &= \text{Rot}(a, \phi_a) \text{Rot}(o, \phi_o) \text{Rot}(n, \phi_n) = \\
 \text{RPY}(\phi_a, \phi_o, \phi_n) &= \text{Rot}(n, \phi_n) \text{Rot}(o, \phi_o) \text{Rot}(a, \phi_a) = \\
 \begin{bmatrix} C\phi_a C\phi_o & C\phi_a S\phi_o S\phi_n - S\phi_a C\phi_n & C\phi_a S\phi_o C\phi_n + S\phi_a S\phi_n & 0 \\ S\phi_a C\phi_o & S\phi_a S\phi_o S\phi_n + C\phi_a C\phi_n & S\phi_a S\phi_o C\phi_n - C\phi_a S\phi_n & 0 \\ -S\phi_o & C\phi_o S\phi_n & C\phi_o C\phi_n & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.36)
 \end{aligned}$$

这个矩阵表示了仅由 RPY 引起的姿态变化。该坐标系相对于参考坐标系的位置和最终姿态是表示位置变化和 RPY 的两个矩阵的乘积。例如,假设一个机器人是根据球坐标和 RPY 来设计的,那么这个机器人就可以表示为:

$${}^R T_H = T_{sph}(r, \beta, \gamma) \times RPY(\phi_a, \phi_o, \phi_n)$$

关于 RPY 的运动学方程的解比球坐标更复杂，因为这里有三个耦合角，所以需要所有三个角各自的正弦和余弦值的信息才能解出这个角。为解出这三个角的正弦值和余弦值，必须将这些角解耦。因此，用 $Rot(a, \phi_a)$ 的逆左乘方程两边，得：

$$Rot(n, \phi_n)^{-1} RPY(\phi_a, \phi_o, \phi_n) = Rot(o, \phi_o) Rot(a, \phi_a) \quad (2.37)$$

假设用 RPY 得到的最后所期望的姿态是用 $(\bar{n}, \bar{o}, \bar{a})$ 矩阵来表示的，则有：

$$Rot(a, \phi_a)^{-1} \begin{bmatrix} n_x & o_x & a_x & 0 \\ n_y & o_y & a_y & 0 \\ n_z & o_z & a_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = Rot(o, \phi_o) Rot(n, \phi_n) \quad (2.38)$$

进行矩阵相乘后得：

$$\begin{bmatrix} n_x C\phi_a + n_y S\phi_a & o_x C\phi_a + o_y S\phi_a & a_x C\phi_a + a_y S\phi_a & 0 \\ n_y C\phi_a - n_x S\phi_a & o_y C\phi_a - o_x S\phi_a & a_y C\phi_a - a_x S\phi_a & 0 \\ n_z & o_z & a_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C\phi_o & S\phi_o S\phi_n & S\phi_o C\phi_n & 0 \\ 0 & C\phi_n & -S\phi_n & 0 \\ -S\phi_o & C\phi_o S\phi_n & C\phi_o C\phi_n & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.39)$$

在式 (2.38) 中的 $\bar{n}, \bar{o}, \bar{a}$ 分量表示了最终的期望值，它们通常是给定或已知的，而 RPY 角的值是未知的变量。

让式 (2.39) 左右两边对应的元素相等，将产生如下结果（关于 ATAN2 函数的解释请参考附录 A）：

根据 2, 1 元素得：

$$n_y C\phi_a - n_x S\phi_a = 0 \rightarrow \phi_a = ATAN2(n_y, n_x) \quad \phi_a = ATAN2(-n_y, -n_x) \quad (2.40)$$

根据 3, 1 元素和 1, 1 元素得：

$$\begin{aligned} S\phi_o &= -n_z \\ C\phi_o &= n_x C\phi_a + n_y S\phi_a \rightarrow \phi_o = ATAN2(-n_z, (n_x C\phi_a + n_y S\phi_a)) \end{aligned} \quad (2.41)$$

根据 2, 2 元素和 2, 3 元素得：

$$\begin{aligned} C\phi_n &= o_y C\phi_a - o_x S\phi_a \\ S\phi_n &= -a_y C\phi_a + a_x S\phi_a \rightarrow \phi_n = ATAN2[(-a_y C\phi_a + a_x S\phi_a), (o_y C\phi_a - o_x S\phi_a)] \end{aligned} \quad (2.42)$$

例 2.16 下面给出了一个笛卡尔坐标-RPY 型机器人手所期望的最终位姿，求滚动角，俯仰角，偏航角和位移。

$${}^R T_P = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.354 & -0.674 & 0.649 & 4.33 \\ 0.505 & 0.722 & 0.475 & 2.50 \\ -0.788 & 0.160 & 0.595 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

解:

根据上述方程, 得到两组解:

$$\phi_a = \text{ATAN2}(n_y, n_x) = \text{ATAN}(0.505, 0.354) = 55^\circ \text{ 或 } 235^\circ$$

$$\phi_o = \text{ATAN2}(-n_z, (n_x C\phi_a + n_y S\phi_a)) = \text{ATAN2}(0.788, 0.616) = 52^\circ \text{ 或 } 128^\circ$$

$$\begin{aligned} \phi_n &= \text{ATAN2}[(-a_y C\phi_a + a_x S\phi_a), (o_y C\phi_a - o_x S\phi_a)] \\ &= \text{ATAN2}(0.259, 0.966) = 15^\circ \text{ 或 } 195^\circ \end{aligned}$$

$$p_x = 4.33 \quad p_y = 2.5 \quad p_z = 8$$

例 2.17 与例 2.16 中的位姿一样, 如果机器人是圆柱坐标-RPY 型, 求所有关节变量。

解:

在这种情况下, 可用:

$${}^R T_P = \begin{bmatrix} 0.354 & -0.674 & 0.649 & 4.33 \\ 0.505 & 0.722 & 0.475 & 2.50 \\ -0.788 & 0.160 & 0.595 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix} = T_{\text{cyl}}(r, \alpha, l) \times \text{RPY}(\Phi_a, \Phi_o, \Phi_n)$$

这个方程右边有 4 个角, 它们是耦合的, 因此必须像前面那样将它们解耦。但是, 因为对于圆柱坐标系 z 轴旋转 Φ_a 角并不影响 a 轴, 所以它仍平行于 z 轴。其结果是, 对于 RPY 绕 a 轴旋转的 α 角可简单地加到 Φ_a 上。这意味着, 求出的 Φ_a 实际上是 $\Phi_a + \alpha$ (参见图 2.23)。根据给定的位置数据, 可求得例 2.16 的解。参考式 (2.33), 得到:

$$rC\alpha = 4.33 \quad rS\alpha = 2.5 \rightarrow \alpha = 30^\circ$$

$$\Phi_a + \alpha = 55^\circ \quad \rightarrow \Phi_a = 25^\circ$$

$$S\alpha = 0.5 \quad \rightarrow r = 5$$

$$p_z = 8 \quad \rightarrow l = 8$$

$$\text{同例 2.16 一样} \quad \rightarrow \Phi_o = 52^\circ \quad \Phi_n = 15^\circ$$

当然, 可以用类似的解法求出第二组解。

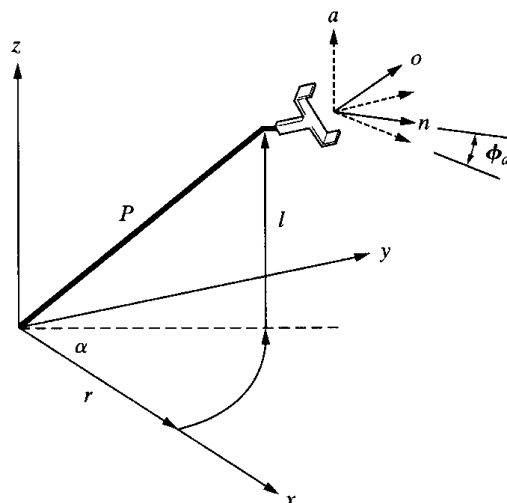


图 2.23 例 2.17 的圆柱和 RPY 坐标

2.7.2.2 欧拉角

除了最后的旋转使然当前的 a 轴外，欧拉角的其他方面均与 RPY 相似（参见图 2.24）。我们仍需要使所有旋转都是绕当前的轴转动以防止机器人的位置有任何改变。表示欧拉角的转动如下：

- 绕 \bar{a} 轴（运动坐标系的 z 轴）旋转 ϕ 度；
- 接着绕 \bar{o} 轴（运动坐标系的 y 轴）旋转 θ 度；
- 最后再绕 \bar{a} 轴（运动坐标系的 x 轴）旋转 ψ 度。

表示欧拉角姿态变化的矩阵是：

$$Euler(\phi, \theta, \psi) = Rot(a, \phi) Rot(o, \theta), Rot(n, \psi) =$$

$$\begin{bmatrix} C\phi C\theta C\psi - S\phi C\psi & -C\phi C\theta S\psi - S\phi C\psi & C\phi S\theta & 0 \\ S\phi C\theta C\psi + C\phi S\psi & -S\phi C\theta S\psi + C\phi C\psi & S\phi S\theta & 0 \\ -S\theta C\psi & S\theta S\psi & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.43)$$

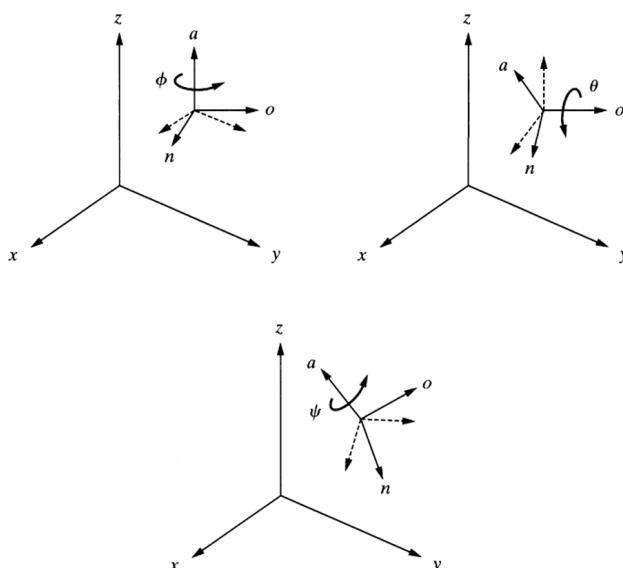


图 2.24 绕当前坐标轴的欧拉旋转

再次强调，该矩阵只是表示了由欧拉角所引起的姿态变化。相对于参考坐标系，这个坐标系的最终位姿是表示位置变化的矩阵和表示欧拉角的矩阵的乘积。

欧拉角的逆运动学求解与 RPY 非常相似。可以使欧拉方程的两边左乘 $Rot^{-1}(a, \phi)$ 来消去其中一边的 ϕ 。让两边的对应元素相等，就可得到以下方程（假设由欧拉角得到的最终所期望的姿态是由 $(\bar{n}, \bar{o}, \bar{a})$ 矩阵表示）：

$$Rot^{-1}(a, \phi) \times \begin{bmatrix} n_x & o_x & a_x & 0 \\ n_y & o_y & a_y & 0 \\ n_z & o_z & a_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C\theta C\psi & -C\theta S\psi & S\theta & 0 \\ S\psi & C\psi & 0 & 0 \\ -S\theta C\psi & S\theta S\psi & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.44)$$

或

$$\begin{bmatrix} n_x C\phi + n_y S\phi & o_x C\phi + o_y S\phi & a_x C\phi + a_y S\phi & 0 \\ -n_x S\phi + n_y C\phi & -o_x S\phi + o_y C\phi & -a_x S\phi + a_y C\phi & 0 \\ n_z & o_z & a_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C\theta C\psi & -C\theta S\psi & S\theta & 0 \\ S\psi & C\psi & 0 & 0 \\ -S\theta C\psi & S\theta S\psi & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.45)$$

记住，式（2.44）中的 $\bar{n}, \bar{o}, \bar{a}$ 表示了最终的期望值，它们通常是给定或已知的。欧拉角的价值是未知变量。

让式（2.45）左右两边对应的元素相等，可得到如下结果：

根据 2, 3 元素，可得：

$$-a_x S\phi + a_y C\phi = 0 \rightarrow \phi = ATAN2(a_y, a_x) \text{ 或 } \phi = ATAN2(-a_y, -a_x) \quad (2.46)$$

由于求得了值，因此式（2.45）左边所有的元素都是已知的。根据 2, 1 元素和 2, 2 元素得：

$$S\psi = -n_x S\phi + n_y C\phi$$

$$C\psi = -o_x S\phi + o_y C\phi \rightarrow \psi = ATAN2(-n_x S\phi + n_y C\phi, -o_x S\phi + o_y C\phi) \quad (2.47)$$

最后根据 1, 3 元素和 3, 3 元素，得：

$$\begin{aligned} S\theta &= a_x C\phi + a_y S\phi \\ C\theta &= a_z \rightarrow \theta = ATAN2(a_x C\phi + a_y S\phi, a_z) \end{aligned} \quad (2.48)$$

例2.18 给定一个直角坐标系—欧拉角型机器人手的最终期望状态，求相应的欧拉角。

$${}^R T_H = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.579 & -0.548 & -0.604 & 5 \\ 0.540 & 0.813 & -0.220 & 7 \\ 0.611 & -0.199 & 0.766 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

解：根据前面的方程，得到：

$$\phi = ATAN2(a_y, a_x) = ATAN2(-0.220, -0.604) = 20^\circ \text{ 或 } 200^\circ$$

将 20° 和 200° 的正弦值和余弦值应用于其他部分，可得：

$$\psi = ATAN2(-n_x S\phi + n_y C\phi, -o_x S\phi + o_y C\phi) = (0.31, 0.952) = 18^\circ \text{ 或 } 198^\circ$$

$$\theta = ATAN2(a_x C\phi + a_y S\phi, a_z) = ATAN2(-0.643, 0.766) = -40^\circ \text{ 或 } 40^\circ$$

2.7.2.3 链式关节

链式关节由 3 个旋转组成，而不是上面刚提出来的旋转模型，这些旋转取决于关节的设计。就像在 2.7.1 (d) 中所做的那样，我们将在讨论 D-H 表示法时来推导表示链式关节的矩阵。

2.7.3 位姿的正逆运动学方程

表示机器人最终位姿的矩阵是前面方程的组合，该矩阵取决于所用的坐标。假设机器人的运动是由直角坐标和 RPY 的组合关节组成，那么该坐标系相对于参考坐标系的最终位姿是表示直角坐标位置变化的矩阵和 RPY 矩阵的乘积。它可表示为：

$${}^R T_H = T_{cart}(P_x, P_y, P_z) \times RPY(\phi_a, \phi_o, \phi_n) \quad (2.49)$$

如果机器人是采用球坐标定位、欧拉角定姿的方式所设计的，那么将得到下列方程。其中位置由球坐标决定，而最终姿态既受球坐标角度的影响也受欧拉角的影响。

$${}^R T_H = T_{sph}(r, \beta, \gamma) \times Euler(\phi, \theta, \psi) \quad (2.50)$$

由于有多种不同的组合，所以这种情况下的正逆运动学解不在此讨论。对于复杂的设计，推荐用 D-H 表示法来求解，并将在下面对此进行讨论。

2.8 机器人正运动学方程的 D-H 表示法

在 1955 年, Denavit 和 Hartenberg 在“ASME Journal of Applied Mechanics”发表了一篇文章, 后来利用那个这篇论文来对机器人进行表示和建模, 并导出了它们的运动方程, 这已成为表示机器人和对机器人运动进行建模的标准方法, 所以必须学习这部分内容。Denavit-Hartenberg(D-H)模型表示了对机器人连杆和关节进行建模的一种非常简单的方法, 可用于任何机器人构型, 而不管机器人的结构顺序和复杂程度如何。它也可用于表示已经讨论过的在任何坐标中的变换, 例如直角坐标、圆柱坐标、球坐标、欧拉角坐标及 RPY 坐标等。另外, 它也可以用于表示全旋转的链式机器人、SCARA 机器人或任何可能的关节和连杆组合。尽管采用前面的方法对机器人直接建模会更快、更直接, 但 D-H 表示法有其附加的好处, 使用它已经开发了许多技术, 例如, 雅克比矩阵的计算和力分析等。

假设机器人由一系列关节和连杆组成。这些关节可能是滑动(线性)的或旋转(转动)的, 它们可以按任意的顺序放置并处于任意的平面。连杆也可以是任意的长度(包括零), 它可能被弯曲或扭曲, 也可能位于任意平面上。所以任何一组关节和连杆都可以构成一个我们想要建模和表示的机器人。

为此, 需要给每个关节指定一个参考坐标系, 然后, 确定从一个关节到下一个关节(一

个坐标系到下一个坐标系)来进行变换的步骤。如果将从基座到第一个关节,再从第一个关节到第二个关节直至到最后一个关节的所有变换结合起来,就得到了机器人的总变换矩阵。在下一节,将根据 D-H 表示法确定一个一般步骤来为每个关节指定参考坐标系,然后确定如何实现任意两个相邻坐标系之间的变换,最后写出机器人的总变换矩阵。

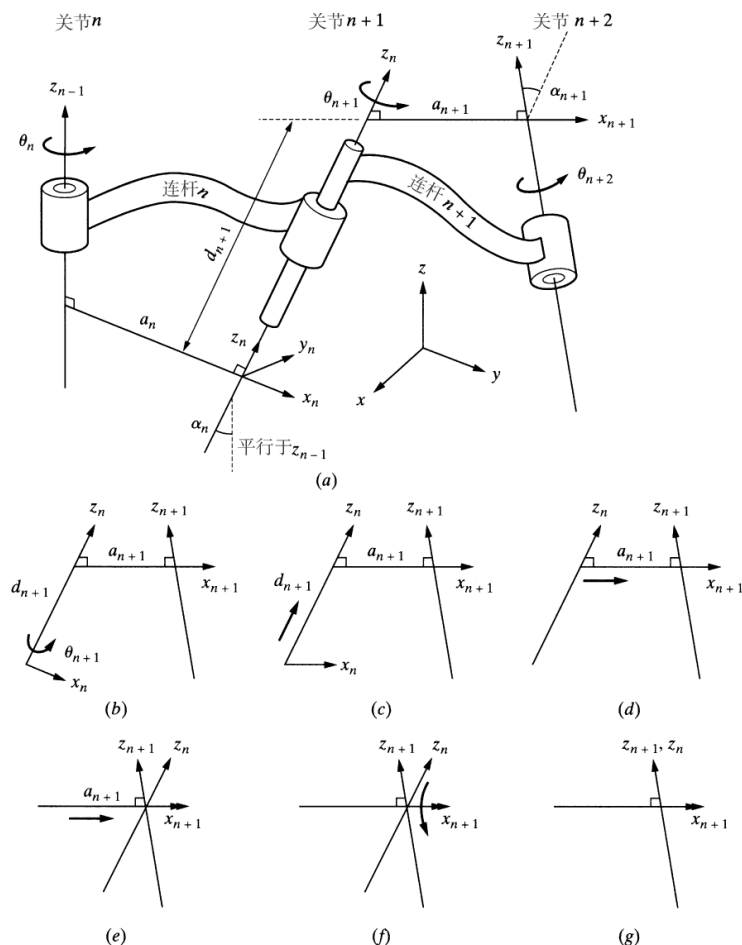


图 2.25 通用关节—连杆组合的 D-H 表示

假设一个机器人由任意多的连杆和关节以任意形式构成。图 2.25 表示了三个顺序的关节和两个连杆。虽然这些关节和连杆并不一定与任何实际机器人的关节或连杆相似,但是他们非常常见,且能很容易地表示实际机器人的任何关节。这些关节可能是旋转的、滑动的、或两者都有。尽管在实际情况下,机器人的关节通常只有一个自由度,但图 2.25 中的关节可以表示一个或两个自由度。

图 2.25 (a) 表示了三个关节,每个关节都是可以转动或平移的。第一个关节指定为关节 n ,第二个关节为关节 $n+1$,第三个关节为关节 $n+2$ 。在这些关节的前后可能还有其他关节。连杆也是如此表示,连杆 n 位于关节 n 与 $n+1$ 之间,连杆 $n+1$ 位于关节 $n+1$ 与 $n+2$ 之间。

为了用 D-H 表示法对机器人建模,所要做的第一件事是为每个关节指定一个本地的参考坐标系。因此,对于每个关节,都必须指定一个 z 轴和 x 轴,通常并不需要指定 y 轴,因为 y 轴总是垂直于 x 轴和 z 轴的。此外, D-H 表示法根本就不用 y 轴。以下是给每个关节指定本地参考坐标系的步骤:

- 所有关节,无一例外的用 z 轴表示。如果关节是旋转的, z 轴位于按右手规则旋转的方向。如果关节是滑动的, z 轴为沿直线运动的方向。在每一种情况下,关节 n 处的 z 轴(以及该关节的本地参考坐标系)的下标为 $n-1$ 。例如,表示关节数 $n+1$ 的 z 轴是 z_n 。这些简单规则可使我们很快地定义出所有关节的 z 轴。对于旋转关节,绕 z 轴的旋转(θ 角)是关节变量。对于滑动关节,沿 z 轴的连杆长度 d 是关节变量。

- 如图 2.25 (a) 所示，通常关节不一定平行或相交。因此，通常 z 轴是斜线，但总有一条距离最短的公垂线，它正交于任意两条斜线。通常在公垂线方向上定义本地参考坐标系的 x 轴。所以如果 a_n 表示 z_{n-1} 与 z_n 之间的公垂线，则 x_n 的方向将沿 a_n 。同样，

在 z_n 与 z_{n+1} 之间的公垂线为 a_{n+1} ， x_{n+1} 的方向将沿 a_{n+1} 。注意相邻关节之间的公垂线不一定相交或共线，因此，两个相邻坐标系原点的位置也可能不在同一个位置。根据上面介绍的知识并考虑下面例外的特殊情况，可以为所有的关节定义坐标系。

- 如果两个关节的 z 轴平行，那么它们之间就有无数条公垂线。这时可挑选与前一关节的公垂线共线的一条公垂线，这样做就可以简化模型。
- 若果两个相邻关节的 z 轴是相交的，那么它们之间就没有公垂线（或者说公垂线距离为零）。这时可将垂直于两条轴线构成的平面的直线定义为 x 轴。也就是说，其公垂线是垂直于包含了两条 z 轴的平面的直线，它也相当于选取两条 z 轴的叉积方向作为 x 轴。这也会使模型得以简化。

在图 2.25 (a) 中， θ 角表示绕 z 轴的旋转角， d 表示在 z 轴上两条相邻的公垂线之间的距离， a 表示每一条公垂线的长度（也叫关节偏移量），角 α 表示两个相邻的 z 轴之间的角度（也叫关节扭转）。通常，只有 θ 和 d 是关节变量。

下一步来完成几个必要的运动，即将一个参考坐标系变换到下一个参考坐标系。假设现在位于本地坐标系 $x_n - z_n$ ，那么通过以下四步标准运动即可到达下一个本地坐标系 $x_{n+1} - z_{n+1}$ 。

- (1) 绕 z_n 轴旋转 θ_{n+1} （如图 2.25 (a) 与 (b) 所示），它使得 x_n 和 x_{n+1} 互相平行，因为 a_n

和 a_{n+1} 都是垂直于 z_n 轴的，因此绕 z_n 轴旋转 θ_{n+1} 使它们平行（并且共面）。

- (2) 沿 z_n 轴平移 d_{n+1} 距离，使得 x_n 和 x_{n+1} 共线（如图 2.25 (c) 所示）。因为 x_n 和 x_{n+1} 已经

平行并且垂直于 z_n ，沿着 z_n 移动则可使它们互相重叠在一起。

- (3) 沿 x_n 轴平移 a_{n+1} 的距离，使得 x_n 和 x_{n+1} 的原点重合（如图 2.25 (d) 和 (e) 所示）。

这时两个参考坐标系的原点处在同一位置。

- (4) 将 z_n 轴绕 x_{n+1} 轴旋转 α_{n+1} ，使得 z_n 轴与 z_{n+1} 轴对准（如图 2.25 (f) 所示）。这时坐标

系 n 和 $n+1$ 完全相同（如图 2.25 (g) 所示）。至此，我们成功地从一个坐标系变换到了下一个坐标系。

在 $n+1$ 和 $n+2$ 坐标系之间严格地按照同样的四个运动顺序可以将一个坐标变换到下一个坐标系。如有必要，可以重复以上步骤，就可以实现一系列相邻坐标系之间的变换。从参考坐标系开始，我们可以将其转换到机器人的基座，然后到第一个关节，第二个关节……，直至末端执行器。这里比较好的一点是，在任何两个坐标系之间的变换均可采用与前面相同的运动步骤。

通过右乘表示四个运动的四个矩阵就可以得到变换矩阵 A ，矩阵 A 表示了四个依次运动。由于所有的变换都是相对于当前坐标系的（即他们都是相对于当前的本地坐标系来测量与执行），因此所有的矩阵都是右乘。从而得到结果如下：

$${}^nT_{n+1} = A_{n+1} = Rot(z, \theta_{n+1}) \times Tran(0, 0, d_{n+1}) \times Tran(a_{n+1}, 0, 0) \times Rot(x, \alpha_{n+1})$$

$$= \begin{bmatrix} C\theta_{n+1} & -S\theta_{n+1} & 0 & 0 \\ S\theta_{n+1} & C\theta_{n+1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_{n+1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.51)$$

$$\times \begin{bmatrix} 1 & 0 & 0 & a_{n+1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_{n+1} & -S\alpha_{n+1} & 0 \\ 0 & S\alpha_{n+1} & C\alpha_{n+1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_{n+1} = \begin{bmatrix} C\theta_{n+1} & -S\theta_{n+1}C\alpha_{n+1} & S\theta_{n+1}S\alpha_{n+1} & a_{n+1}C\theta_{n+1} \\ S\theta_{n+1} & C\theta_{n+1}C\alpha_{n+1} & -C\theta_{n+1}S\alpha_{n+1} & a_{n+1}S\theta_{n+1} \\ 0 & S\alpha_{n+1} & C\alpha_{n+1} & d_{n+1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.52)$$

比如，一般机器人的关节 2 与关节 3 之间的变换可以简化为：

$${}^2T_3 = A_3 = \begin{bmatrix} C\theta_3 & -S\theta_3C\alpha_3 & S\theta_3S\alpha_3 & a_3C\theta_3 \\ S\theta_3 & C\theta_3C\alpha_3 & -C\theta_3S\alpha_3 & a_3S\theta_3 \\ 0 & S\alpha_3 & C\alpha_3 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.53)$$

在机器人的基座上，可以从第一个关节开始变换到第二个关节，然后到第三个……，再到机器人的手，最终到末端执行器。若把每个变换定义为，则可以得到许多表示变换的矩阵。在机器人的基座与手之间的总变换则为：

$${}^RT_H = {}^RT_1 {}^1T_2 {}^2T_3 \cdots {}^{n-1}T_n = A_1 A_2 A_3 \cdots A_n \quad (2.54)$$

其中 n 是关节数。对于一个具有六个自由度的机器人而言，有 6 个 A 矩阵。

为了简化 A 矩阵的计算，可以制作一张关节和连杆参数的表格，其中每个连杆和关节的参数值可从机器人的原理示意图上确定，并且可将这些参数代入 A 矩阵。表 2.1 可用于这个目的。

在以下几个例子中，我们将建立必要的坐标系，填写参数表，并将这些数值代入 A 矩阵。首先从简单的机器人开始，以后再考虑复杂的机器人。

表 2.1 D-H 参数表

#	θ	d	a	α
1				
2				
3				
4				
5				
6				

例2.19 对于如图 2.26 所示的简单机器人，根据 D-H 表示法，建立必要的坐标系，并填写相应的参数表。

解：

为方便起见，在此例中，假设关节 2, 3 和 4 在同一平面内，即它们的 d_n 值为 0。为建

立机器人的坐标系，首先寻找关节（如图 2.26 所示）。该机器人有六个自由度，在这个简单机器人中，所有的关节都是旋转的。第一个关节（关节 1）在连杆 0（固定基座）和连杆 1 之间，关节 2 在连杆 1 和连杆 2 之间，等等。首先，如前面已经讨论过的那样，对每个关节建立 z 轴，接着建立 z 轴。观察图 2.27 和图 2.28 所示的坐标可以发现，图 2.28 是图 2.27 的简化线图。应注意每个坐标系原点 3 在它所在位置的原因。

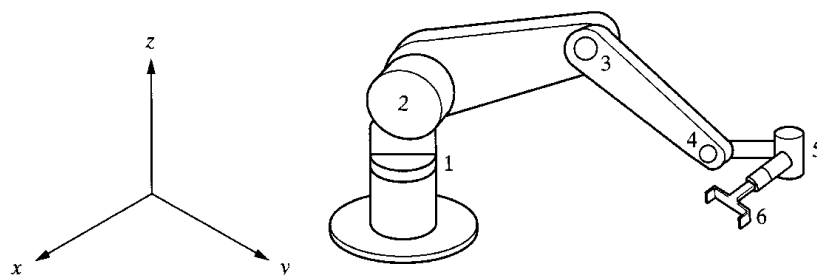


图 2.26 具有六个自由度的简单链式机器人

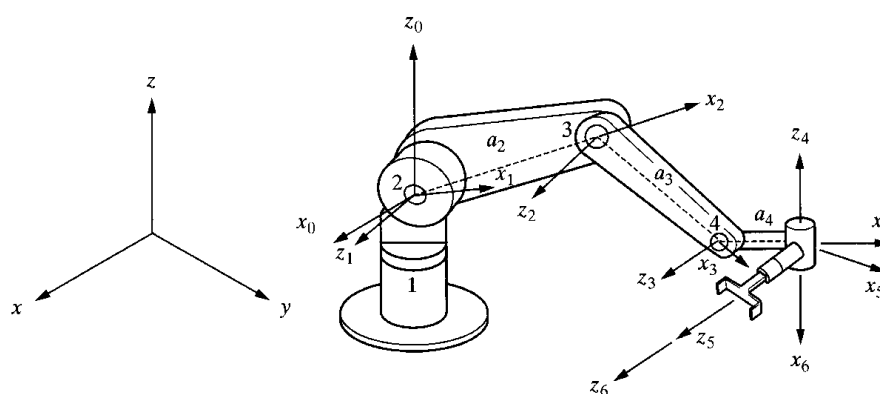


图 2.27 简单六个自由度链式机器人的参考坐标系

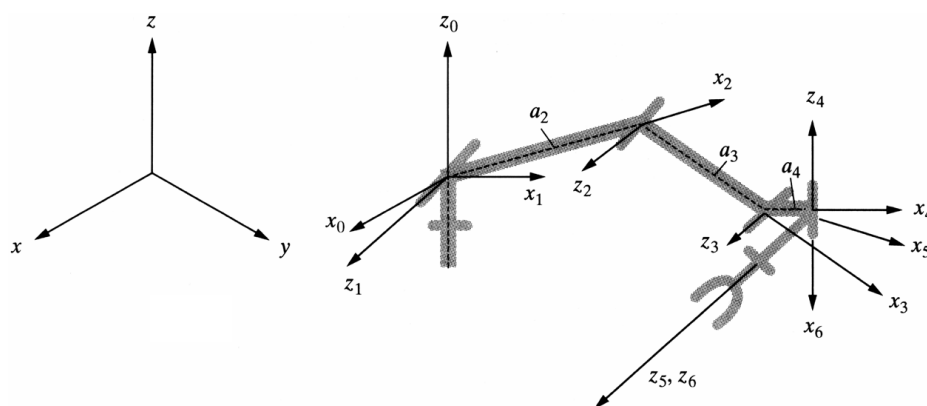


图 2.28 简单六个自由度链式机器人的参考坐标系线图

从关节 1 开始， z_0 表示第一个关节，它是一个旋转关节。选择 x_0 与参考坐标系的 x 轴平行，这样做仅仅是为了方便， x_0 是一个固定的坐标轴，表示机器人的基座，它是不动的。第一个关节的运动是围绕着 z_0 - x_0 轴进行的，但这两个轴并不运动。接下来，在关节 2 处设定 z_1 ，因为坐标轴 z_0 和 z_1 是相交的，所以 x_1 垂直于 z_0 和 z_1 。 x_2 在 z_1 和 z_2 之间的公垂线方向上， x_3 在 z_2 和 z_3 之间的公垂线方向上，类似地， x_4 在 z_3 和 z_4 之间的公垂线方向上。最后， z_5 和 z_6 是平

行且共线的。 z_5 表示关节 6 的运动，而 z_6 表示末端执行的运动。通常在运动方程中不包含末端执行器，但应包含末端执行器的坐标系，这是因为它可以容许进行从坐标系 $z_5 - x_5$ 出发的变换。同时也要注意第一个和最后一个坐标系的原点的位置，它们将决定机器人的总变换方程。可以在第一个和最后的坐标系之间建立其他的（或不同的）中间坐标系，但只要第一个和最后的坐标系没有改变，机器人的总变换便是不变的。应注意的是，第一个关节的原点并不在关节的实际位置，但证明这样做是没有问题的，因为无论实际关节是高一点还是低一点，机器人的运动并不会有任何差异。因此，考虑原点位置时可不用考虑基座上关节的实际位置。

接下来，我们将根据已建立的坐标系来填写表 2.2 中的参数。参考前一节中任意两个坐标系之间的四个运动的顺序。从 $z_0 - x_0$ 开始，有一个旋转运动将 x_0 转到了 x_1 ，为使得 x_0 与 x_1 轴重合，需要沿 z_1 和沿 x_1 的平移均为零，还需要一个旋转将 z_0 转到 z_1 ，注意旋转是根据右手规则进行的，即将右手手指按旋转的方向弯曲，大拇指的方向则为旋转坐标轴的方向。到了这时， $z_0 - x_0$ 就变换到了 $z_1 - x_1$ 。

接下来，绕 z_1 旋转 θ_2 ，将 x_1 转到了 x_2 ，然后沿 x_2 轴移动距离 a_2 ，使坐标系原点重合。由于前后两个 z 轴是平行的，所以没有必要绕 x 轴旋转。按照这样的步骤继续做下去，就能得到所需要的结果。

必须要认识到，与其他机械类似，机器人也不会保持原理图中所示的一种构型不变。尽管机器人的原理图是二维的，但必须要想象出机器人的运动，也就是说，机器人的不同连杆和关节在运动时，与之相连的坐标系也随之运动。如果这时原理图所示机器人构型的坐标轴处于特殊的位姿状态，当机器人移动时它们又会处于其他的点和姿态上。比如， x_3 总是沿着关节 3 与关节 4 之间连线 a_3 的方向。当机器人的下臂绕关节 2 旋转而运动。在确定参数时，必须记住这一点。

表 2.2 例 2.19 机器人的参数

#	θ	d	a	α
1	θ_1	0	0	90
2	θ_2	0	a_2	0
3	θ_3	0	a_3	0
4	θ_4	0	a_4	-90
5	θ_5	0	0	90
6	θ_6	0	0	0

θ 表示旋转关节的关节变量，d 表示滑动关节的关节变量。因为这个机器人的关节全是旋转的，因此所有关节变量都是角度。

通过简单地从参数表中选取参数代入 A 矩阵，便可写出每两个相邻关节之间的变换。例

如,在坐标系 0 和 1 之间的变换矩阵 A_1 可通过将 α ($\sin 90^\circ=1, \cos 90^\circ=0, \alpha=90^\circ$) 以及指定 C_1 为 θ_1 等代入 A 矩阵得到, 对其他关节的 $A_2 \sim A_6$ 矩阵也是这样, 最后得:

$$\begin{aligned}
 A_1 &= \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & A_2 &= \begin{bmatrix} C_2 & -S_2 & 0 & C_2 a_2 \\ S_2 & C_2 & 0 & S_2 a_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 A_3 &= \begin{bmatrix} C_3 & -S_3 & 0 & C_3 a_3 \\ S_3 & C_3 & 0 & S_3 a_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & A_4 &= \begin{bmatrix} C_4 & 0 & -S_4 & C_4 a_4 \\ S_4 & 0 & C_4 & S_4 a_4 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 A_5 &= \begin{bmatrix} C_5 & 0 & S_5 & 0 \\ S_5 & 0 & -C_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & A_6 &= \begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned} \tag{2.55}$$

特别注意: 为简化最后的解, 将用到下列三角函数关系式:

$$\begin{aligned}
 S\theta_1 C\theta_2 + C\theta_1 S\theta_2 &= S(\theta_1 + \theta_2) = S_{12} \\
 C\theta_1 C\theta_2 - S\theta_1 S\theta_2 &= C(\theta_1 + \theta_2) = C_{12}
 \end{aligned} \tag{2.56}$$

在机器人的基座和手之间的总变换为:

$${}^R T_H = A_1 A_2 A_3 A_4 A_5 A_6 \tag{2.57}$$

$$= \begin{bmatrix} C_1(C_{234}C_5C_6 - S_{234}S_6) & C_1(-C_{234}C_5C_6 - S_{234}C_6) & C_1(C_{234}S_5) & C_1(C_{234}a_4 + \\ -S_1S_5C_6 & +S_1S_5S_6 & +S_1C_5 & C_{23}a_3 + C_2a_2) \\ S_1(C_{234}C_5C_6 - S_{234}S_6) & S_1(-C_{234}C_5C_6 - S_{234}C_6) & S_1(C_{234}S_5) & S_1(C_{234}a_4 + \\ +C_1S_5S_6 & -C_1S_5S_6 & -C_1C_5 & C_{23}a_3 + C_2a_2) \\ S_{234}C_5C_6 & -S_{234}C_5C_6 + C_{234}C_6 & S_{234}S_5 & S_{234}a_4 + S_{23}a_3 + S_2a_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

例2.20 斯坦福机械手臂。在斯坦福机械手臂上指定坐标系(如图 2.29 所示), 并填写参数表。斯坦福机械手臂是一个球坐标手臂, 即开始的两个关节是旋转的, 第三个关节是滑动的, 最后三个腕关节全是旋转关节。

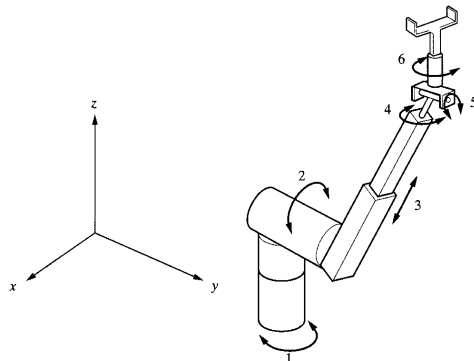


图 2.29 斯坦福机械手臂示意图

解:

在看本题解答之前, 现根据自己的理解来做, 问题的答案在本章的最后。建议在看解答中建立的坐标系和机械手臂的解之前, 先试着自己做。

机器手臂最后的正运动学解是相邻关节之间的 6 个变换矩阵的乘积:

$${}^R T_{H_{STANDARD}} = {}^0 T_6 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

其中

$$\begin{aligned} n_x &= C_1[C_2(C_4C_5C_6 - S_4S_6) - S_2S_5C_6] - S_1(S_4C_5C_6 + C_4S_6) \\ n_y &= S_1[C_2(C_4C_5C_6 - S_4S_6) - S_2S_5C_6] + C_1(S_4C_5C_6 + C_4S_6) \\ n_z &= -S_2(C_4C_5C_6 - S_4S_6) - C_2S_5C_6 \\ o_x &= C_1[-C_2(C_4C_5C_6 + S_4S_6) + S_2S_5C_6] - S_1(-S_4C_5C_6 + C_4S_6) \\ o_y &= S_1[-C_2(C_4C_5C_6 + S_4S_6) + S_2S_5C_6] + C_1(-S_4C_5C_6 + C_4S_6) \\ o_z &= S_2(C_4C_5C_6 + S_4S_6) + C_2S_5C_6 \\ a_x &= C_1(C_2C_4S_5 + S_2C_6) - S_1S_4S_5 \\ a_y &= S_1(C_2C_4S_5 + S_2C_6) + C_1S_4S_5 \\ a_z &= -S_2C_4S_5 + C_2C_5 \\ p_x &= C_1S_2d_3 - S_1d_2 \\ p_y &= S_1S_2d_3 + C_1d_2 \\ p_z &= C_2d_3 \end{aligned} \quad (2.58)$$

2.9 机器人的逆运动学解

如前所述, 这里真正重要的是逆运动学解。为了使机器人手臂处于期望的位姿, 如果有逆运动学解就能确定每个关节的值。前面已对特定坐标系统的逆运动学解作了介绍。在这一部分, 将研究求解逆运动方程的一般步骤。

现在你可能已经注意到, 前面的运动方程中有许多角度的耦合, 比如 C_{234} , 这就使得无法从矩阵中提取足够的元素来求解单个的正弦和余弦项以计算角度。为使角度解耦, 可例行地用单个 ${}^R T_H$ 矩阵左乘 A_n^{-1} 矩阵, 使得方程右边不再包括这个角度, 于是可以找到产生角度的正弦值和余弦值的元素, 并进而求得相应的角度。

这里概要地给出了这个方法, 并将其用于例 2.19 中的简单机械手臂。虽然所给出的解决方法只针对这一给定构型的机器人, 但也可以类似地用于其它机器人。正如在例 2.19 中看到的, 表示机器人的最后方程为:

$${}^R T_H = A_1 A_2 A_3 A_4 A_5 A_6$$

$$= \begin{bmatrix} C_1(C_{234}C_5C_6 - S_{234}S_6) & C_1(-C_{234}C_5C_6 - S_{234}C_6) & C_1(C_{234}S_5) & C_1(C_{234}a_4 + \\ -S_1S_5C_6 & +S_1S_5S_6 & +S_1C_5 & C_{23}a_3 + C_2a_2) \\ S_1(C_{234}C_5C_6 - S_{234}S_6) & S_1(-C_{234}C_5C_6 - S_{234}C_6) & S_1(C_{234}S_5) & S_1(C_{234}a_4 + \\ +C_1S_5S_6 & -C_1S_5S_6 & -C_1C_5 & C_{23}a_3 + C_2a_2) \\ S_{234}C_5C_6 + C_{234}C_6 & -S_{234}C_5C_6 + C_{234}C_6 & S_{234}S_5 & S_{234}a_4 + S_{23}a_3 + S_2a_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

为了书写方便，将上面的矩阵表示为[RHS](Right-Hand Side)。这里再次将机器人的期望位姿表示为：

$${}^R T_H = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.59)$$

为了求解角度，从 A_n^{-1} 开始，依次用 A_1^{-1} 左乘上述两个矩阵，得到：

$$A_1^{-1} \times \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = A_1^{-1}[\text{RHS}] = A_2 A_3 A_4 A_5 A_6 \quad (2.60)$$

$$\begin{bmatrix} C_1 & S_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ S_1 & -C_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = A_2 A_3 A_4 A_5 A_6$$

$$\begin{bmatrix} n_x C_1 + n_y S_1 & o_x C_1 + o_y S_1 & a_x C_1 + a_y S_1 & p_x C_1 + p_y S_1 \\ n_z & o_z & a_z & p_z \\ n_x S_1 - n_y C_1 & o_x S_1 - o_y C_1 & a_x S_1 - a_y C_1 & p_x S_1 - p_y C_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} C_{234}C_5C_6 - S_{234}S_6 & -C_{234}C_5C_6 - S_{234}C_6 & C_{234}S_5 & C_{234}a_4 + C_{23}a_3 + C_2a_2 \\ S_{234}C_5C_6 + C_{234}S_6 & -S_{234}C_5C_6 + C_{234}C_6 & S_{234}S_5 & S_{234}a_4 + S_{23}a_3 + S_2a_2 \\ -S_5C_6 & S_5S_6 & C_5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.61)$$

根据方程的 3, 4 元素，有：

$$p_x S_1 - p_y C_1 = 0 \quad \rightarrow \quad \theta_1 = \arctan\left(\frac{p_y}{p_x}\right) \text{ 和 } \theta_1 = \theta_1 + 180^\circ \quad (2.62)$$

根据 1, 4 元素和 2, 4 元素，可得：

$$p_x C_1 + p_y S_1 = C_{234}a_4 + C_{23}a_3 + C_2a_2$$

$$p_z = S_{234}a_4 + S_{23}a_3 + S_2a_2 \quad (2.63)$$

整理上面两个方程并对两边平方，然后将平方值相加，得：

$$\begin{aligned}(p_x C_1 + p_y S_1 - C_{234} a_4)^2 &= (C_{23} a_3 + C_2 a_2)^2 \\ (p_z - S_{234} a_4)^2 &= (S_{23} a_3 + S_2 a_2)^2\end{aligned}$$

$$(p_x C_1 + p_y S_1 - C_{234} a_4)^2 + (p_z - S_{234} a_4)^2 = a_2^2 + a_3^2 + 2a_2 a_3 (S_2 S_{23} + C_2 C_{23})$$

根据式 (2.56) 的三角函数方程, 可得:

$$S_2 S_{23} + C_2 C_{23} = \cos[(\theta_2 + \theta_3) - \theta_2] = \cos \theta_3$$

于是:

$$C_3 = \frac{(p_x C_1 + p_y S_1 - C_{234} a_4)^2 + (p_z - S_{234} a_4)^2 - a_2^2 - a_3^2}{2a_2 a_3} \quad (2.64)$$

在这个方程中, 除 S_{234} 和 C_{234} 外, 每个变量都是已知的, S_{234} 和 C_{234} 将在后面求出。已知:

$$S_3 = \pm \sqrt{1 - C_3^2}$$

于是可得:

$$\theta_3 = \arctan \frac{S_3}{C_3} \quad (2.65)$$

因为关节 2, 3 和 4 都是平行的, 左乘 A_2 和 A_3 的逆不会产生有用的结果。下一步左乘 $A_1 \sim A_4$ 的逆, 结果为:

$$A_4^{-1} A_3^{-1} A_2^{-1} A_1^{-1} \times \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = A_4^{-1} A_3^{-1} A_2^{-1} A_1^{-1} [\text{RHS}] = A_5 A_6 \quad (2.66)$$

乘开后可得:

$$\begin{aligned} & \begin{bmatrix} C_{234}(C_1 n_x + S_1 n_y) + S_{234} n_z & C_{234}(C_1 o_x + S_1 o_y) + S_{234} o_z & C_{234}(C_1 a_x + S_1 a_y) + S_{234} a_z & C_{234}(C_1 p_x + S_1 p_y) + S_{234} p_z \\ C_1 n_y - S_1 n_x & C_1 o_y - S_1 o_x & C_1 a_y - S_1 a_x & 0 \\ -S_{234}(C_1 n_x + S_1 n_y) + C_{234} n_z & -S_{234}(C_1 o_x + S_1 o_y) + C_{234} o_z & -S_{234}(C_1 a_x + S_1 a_y) + C_{234} a_z & -S_{234}(C_1 p_x + S_1 p_y) + C_{234} p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} C_5 C_6 & -C_5 S_6 & S_5 & 0 \\ S_5 C_6 & -S_5 S_6 & -C_5 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (2.67)$$

根据式 (2.67) 矩阵的 3, 3 元素,

$$-S_{234}(C_1 a_x + S_1 a_y) + C_{234} a_z = 0 \quad \rightarrow$$

$$\theta_{234} = \arctan\left(\frac{a_z}{C_1 a_x + S_1 a_y}\right) \text{ 和 } \theta_{234} = \theta_{234} + 180^\circ \quad (2.68)$$

由此可计算 S_{234} 和 C_{234} ，如前面所讨论过的，它们可用来计算 θ_3 。

现在再参照式 (2.63)，并在这里重复使用它就可计算角 θ_2 的正弦和余弦值。具体步骤如下：

$$\begin{cases} p_x C_1 + p_y S_1 = C_{234} a_4 + C_{23} a_3 + C_2 a_2 \\ p_z = S_{234} a_4 + S_{23} a_3 + S_2 a_2 \end{cases}$$

由于 $C_{12} = C_1 C_2 - S_1 S_2$ 以及 $S_{12} = S_1 C_2 + C_1 S_2$ ，可得：

$$\begin{cases} p_x C_1 + p_y S_1 - C_{234} a_4 = (C_2 C_3 - S_2 S_3) a_3 + C_2 a_2 \\ p_z - S_{234} a_4 = (S_2 C_3 + C_2 S_3) a_3 + S_2 a_2 \end{cases} \quad (2.69)$$

上面两个方程中包含两个未知数，求解 C_2 和 S_2 ，可得：

$$\begin{cases} S_2 = \frac{(C_3 a_3 + a_2)(p_z - S_{234} a_4) - S_3 a_3 (p_x C_1 + p_y S_1 - C_{234} a_4)}{(C_3 a_3 + a_2)^2 + S_3^2 a_3^2} \\ C_2 = \frac{(C_3 a_3 + a_2)(p_x C_1 + p_y S_1 - C_{234} a_4) + S_3 a_3 (p_z - S_{234} a_4)}{(C_3 a_3 + a_2)^2 + S_3^2 a_3^2} \end{cases} \quad (2.70)$$

尽管这个方程较复杂，但它的所有元素都是已知的，因此可以计算得到：

$$\theta_2 = \arctan \frac{(C_3 a_3 + a_2)(p_z - S_{234} a_4) - S_3 a_3 (p_x C_1 + p_y S_1 - C_{234} a_4)}{(C_3 a_3 + a_2)(p_x C_1 + p_y S_1 - C_{234} a_4) + S_3 a_3 (p_z - S_{234} a_4)} \quad (2.71)$$

既然 θ_2 和 θ_3 已知，进而可得：

$$\theta_4 = \theta_{234} - \theta_2 - \theta_3 \quad (2.72)$$

因为式 (2.68) 中的 θ_{234} 有两个解，所以 θ_4 也有两个解。

根据式 (2.67) 中的 1, 3 元素和 2, 3 元素，可以得到：

$$\begin{cases} S_5 = C_{234}(C_1 a_x + S_1 a_y) + S_{234} a_z \\ C_5 = -C_1 a_y + S_1 a_x \end{cases} \quad (2.73)$$

和

$$\theta_5 = \arctan \frac{C_{234}(C_1 a_x + S_1 a_y) + S_{234} a_z}{S_1 a_x - C_1 a_y} \quad (2.74)$$

也许已注意到，因为对于 θ_6 没有解耦方程，所以必须用 A_5 矩阵的逆左乘式 (2.67) 来对它解耦。这样做后可得到：

$$\begin{bmatrix} C_5 [C_{234}(C_1 n_x + S_1 n_y) + S_{234} n_z] & C_5 [C_{234}(C_1 o_x + S_1 o_y) + S_{234} o_z] & 0 & 0 \\ -S_5 (S_1 n_x - C_1 n_y) & -S_5 (S_1 o_x - C_1 o_y) & 0 & 0 \\ -S_{234}(C_1 n_x + S_1 n_y) + C_{234} n_z & -S_{234}(C_1 o_x + S_1 o_y) + C_{234} o_z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.75)$$

根据式 (2.75) 中的 2, 1 元素和 2, 2 元素, 得到:

$$\theta_6 = \arctan \frac{-S_{234}(C_1 n_x + S_1 n_y) + C_{234} n_z}{-S_{234}(C_1 o_x + S_1 o_y) + C_{234} o_z} \quad (2.76)$$

至此找到了 6 个方程, 它们合在一起给出了机器人置于任何期望位姿时所需的关节值。虽然这种方法仅适用于给定的机器人, 但也可采取类似的方法来处理其他的机器人。

值得注意的是, 仅仅因为机器人的最后三个关节交于一个公共点才使得这个方法有可能求解, 否则就不能用这个方法求解, 而只能直接求解矩阵或通过计算矩阵的逆来求解未知的量。大多数工业机器人都有相交的腕关节。

2.10 机器人的逆运动学编程

求解机器人逆运动问题所建立的方程可以直接用于驱动机器人到达一个位置。事实上, 没有机器人真正用正运动方程求解这个问题, 所用到的仅为计算关节值的 6 个方程, 并反过来用它们驱动机器人到达期望位置。这样做是必须的, 其实际原因是: 计算机计算正运动方程的逆或将值代入正运动方程, 并用高斯消去法来求解未知量 (关节变量) 将花费大量时间。

为使机器人按预定的轨迹运动, 譬如说直线, 那么在一秒内必须多次反复计算关节变量。现假设机器人沿直线从起点 A 运动到终点 B, 如果期间不采取其他措施, 那么机器人从 A 运动到 B 的轨迹难以预测。机器人将运动它的所有关节直到他们都到达终值, 这是机器人便到达了终点 B, 然而, 机器人手在两点间运行的路径是未知的, 它取决于机器人每个关节的变化率。为了使机器人按直线运动, 必须把这一路径分成如图 2.30 所示的许多小段, 让机器人按照分好的小段路径在两点间依次运动。这就意味着对每一小段路径都必须计算新的逆运动学解。典型情况下, 每秒钟要对位置反复计算 50~200 次。也就是说, 如果计算逆解耗时 5~20ms 以上, 那么机器人将丢失精度或不能按照指定路径运动。用来计算新解的时间越短, 机器人的运动据越精确。因此, 必须尽量减少不必要的计算, 从而使计算机控制器能做更多的逆解计算。这也就是为什么设计者必须事先做好所有的数学处理, 并仅需为计算机控制器编程来计算最终的解的原因。第 5 章将详细讨论这个问题。

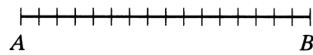


图 2.30 直线的小段运动

对于早先讨论过的旋转机器人情况, 给定最终的期望位姿为:

$${}^R T_{HDESIRED} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

为了计算未知角度, 控制器所需要的所有计算是如下的一组逆解:

$$\begin{aligned}
\theta_1 &= \arctan\left(\frac{p_y}{p_x}\right) \text{ 和 } \theta_1 = \theta_1 + 180^\circ \\
\theta_{234} &= \arctan\left(\frac{a_z}{C_1 a_x + S_1 a_y}\right) \text{ 和 } \theta_{234} = \theta_{234} + 180^\circ \\
C_3 &= \frac{(p_x C_1 + p_y S_1 - C_{234} a_4)^2 + (p_z - S_{234} a_4)^2 - a_2^2 - a_3^2}{2a_2 a_3} \\
S_3 &= \pm \sqrt{1 - C_3^2} \\
\theta_3 &= \arctan \frac{S_3}{C_3} \\
\theta_2 &= \arctan \frac{(C_3 a_3 + a_2)(p_z - S_{234} a_4) - S_3 a_3 (p_x C_1 + p_y S_1 - C_{234} a_4)}{(C_3 a_3 + a_2)(p_x C_1 + p_y S_1 - C_{234} a_4) + S_3 a_3 (p_z - S_{234} a_4)} \\
\theta_4 &= \theta_{234} - \theta_2 - \theta_3 \\
\theta_5 &= \arctan \frac{C_{234} (C_1 a_x + S_1 a_y) + S_{234} a_z}{S_1 a_x - C_1 a_y} \\
\theta_6 &= \arctan \frac{-S_{234} (C_1 n_x + S_1 n_y) + C_{234} n_z}{-S_{234} (C_1 o_x + S_1 o_y) + C_{234} o_z} \quad (2.77)
\end{aligned}$$

虽然以上计算也并不简单，但用这些方程来计算角度要比对矩阵求逆或使用高斯消去法计算要快得多。这里所有的运算都是简单的算术运算和三角运算。

2.11 机器人的退化和灵巧特性

退化 当机器人失去一个自由度，并因此不按所期望的状态运动时即称机器人发生了退化。在两种条件下会发生退化：（1）机器人关节达到其物理极限而不能进一步运动，（2）如果两个相似关节的 z 轴共线时，机器人可能会在其工作空间中变为退化状态。这意味此时无论哪个关节运动都将产生同样的运动，结果是控制器将不知道是哪个关节在运动。无论哪一种情况，机器人的自由度总数都小于 6，因此机器人的方程无解。在关节共线时，位置矩阵的行列式也为零。图 2.31 显示了一个处于垂直构型的简单机器人，其中关节 1 和 6 共线。可以看到，无论关节 1 或关节 6 旋转，末端执行器的旋转结果都是一样的。实际上，这时指令控制器采取紧急行动是十分重要的，否则机器人将停止运行。应注意，这种情况只在两关节相似时才会发生，反之，如果一个关节是滑动型的，而另一个是旋转型的（例如斯坦福机械手臂的关节 3 和关节 4），那么即使它们的 z 轴共线，机器人也不会出现退化的现象。

Paul 指出：如果 $\sin(\alpha_4), \sin(\alpha_5)$, 或 $\sin(\theta_5)$ 为 0，机器人就将退化。显然，可以适当设计 α_4 和 α_5 来防止机器人退化。此外，如果任何时候 θ_5 接近 0° 或者 180° ，机器人就将变成退化状态。

灵巧 一般认为只要确定了机器人手的位姿，就能为具有六个自由度的机器人在其工作范围内的任何位置定位和定姿。实际上，随着机器人越来越接近其工作空间的极限，虽然机器人仍可能定位在期望的点上，但却有可能不能定姿在期望的位姿上。能对机器人定位但不能对它定姿的点的区域成为不灵巧区域。

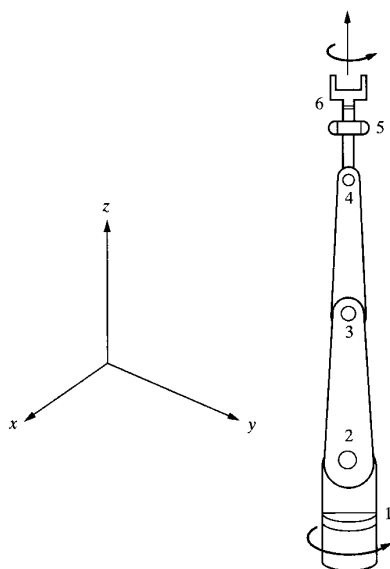


图 2.31 处于退化状态的机器人

2.12 D-H 表示法的基本问题

虽然 D-H 表示法已广泛用于机器人的运动建模和分析，并已成为解决该问题的标准方法，但它在技术上仍存在着根本的缺陷，很多研究者试图通过改进 D-H 表示法来解决这个问题。其根本问题是：由于所有的运动都是关于 x 和 z 轴的，而无法表示关于 y 轴的运动，因此只要有任何关于 y 轴的运动，此方法就不适用，而且这种情况十分普遍。例如，假设原本应该平行的两个关节轴在安装时有一点小的偏差，由于两轴之间存在小的夹角，因此需要沿 y 轴运动。由于实际的工业机器人在其制造过程中都存在一定的误差，所以该误差不能用 D-H 法来建模。

例 2.20 (续) 斯坦福机械手臂的参考坐标系。图 2.32 显示了例 2.20 中（如图 2.29 所示）斯坦福机械手臂的参考坐标系。为了改进可视性，进行了简化。表 2.3 所示为斯坦福机械手臂的参数表。

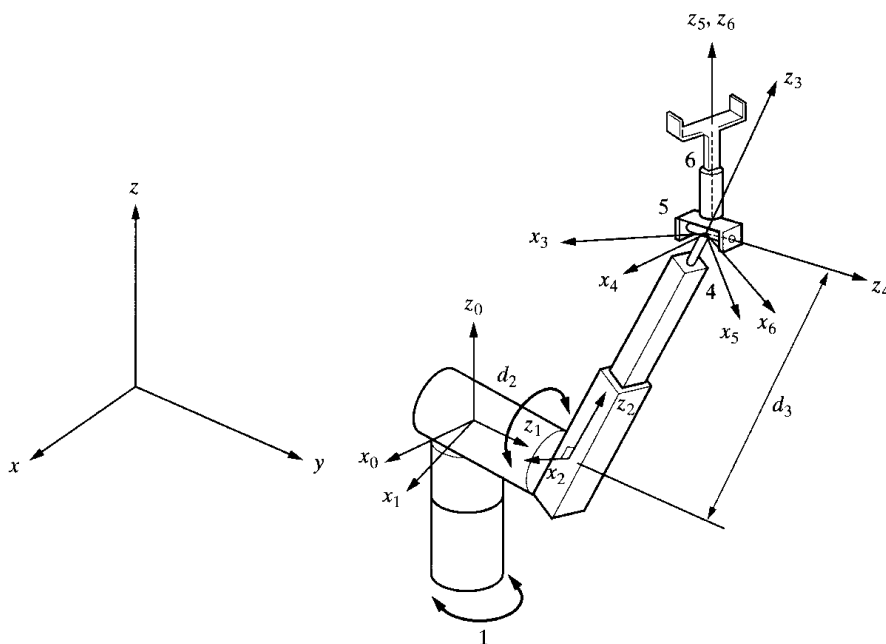


图 2.32 斯坦福机械手臂的坐标系

关于斯坦福机械手臂逆运动解的推导，请见参考文献[5,13]。以下是斯坦福手臂逆运动学解的结果汇总：

$$\theta_1 = \arctan\left(\frac{p_y}{p_x}\right) - \arctan\frac{d_2}{\pm\sqrt{r^2 - d_2^2}} \quad \text{其中} \quad r = \sqrt{p_x^2 + p_y^2} \quad (2.78)$$

$$\theta_2 = \arctan\frac{C_1 p_x + S_1 p_y}{p_z} \quad (2.79)$$

$$d_3 = S_2(C_1 p_x + S_1 p_y) + C_2 p_z \quad (2.80)$$

$$\theta_4 = \arctan\left(\frac{-S_1 a_x + C_1 a_y}{C_2(C_1 a_x + S_1 a_y) - S_2 a_z}\right) \text{和} \theta_4 = \theta_4 + 180^\circ \text{ 如果 } \theta_5 < 0 \quad (2.81)$$

$$\theta_5 = \arctan\frac{C_4[C_2(C_1 a_x + S_1 a_y) - S_2 a_z] + S_4[-S_1 a_x + C_1 a_y]}{S_2(C_1 a_x + S_1 a_y) + C_2 a_z} \quad (2.82)$$

$$\theta_6 = \arctan\frac{S_6}{C_6} \quad \text{其中}$$

$$S_6 = -C_5\{C_4[C_2(C_1 o_x + S_1 o_y) - S_2 o_z] + S_4[-S_1 o_x + C_1 o_y]\} \\ + S_5\{S_2(C_1 o_x + S_1 o_y) + C_2 o_z\} \quad (2.83)$$

$$C_6 = -S_4[C_2(C_1 o_x + S_1 o_y) - S_2 o_z] + C_4[-S_1 o_x + C_1 o_y]$$

表 2.3 图 2.32 所示斯坦福机械手臂的参数表

#	θ	d	a	α
1	θ_1	0	0	-90
2	θ_2	d_2	0	90
3	0	d_3	0	0
4	θ_4	0	0	-90
5	θ_5	0	0	90
6	θ_6	0	0	0

2.13 设计项目：四自由度机器人

利用本书中所介绍的四自由度机器人，结合本章所学的知识进行四自由度机器人的正逆运动学分析。

SCARA 型机器人的运动学模型的建立，包括机器人运动学方程的表示，以及运动学正解、逆解等，这些是研究机器人控制的重要基础，也是开放式机器人系统轨迹规划的重要基础。为了描述 SCARA 型机器人各连杆之间的数学关系，在此采用 Denavit 和 Hertenberg 提出的齐次变换矩阵的方法，即 D-H 法。SCARA 型机器人操作臂可以看作是一个开式运动链。它是由一系列连杆通过转动或移动关节串联而成的。为了研究操作臂各连杆之间的位移关系，可

在每个连杆上固接一个坐标系，然后描述这些坐标系之间的关系。

2.13.1 SCARA 机器人坐标系的建立

1.SCARA 机器人坐标系建立原则根据 D-H 坐标系建立方法，SCARA 机器人的每个关节坐标系的建立可参照以下的三原则

(1) z_n 轴沿着第 n 个关节的运动轴;基坐标系的选择为:当第一关节变量为零时，零坐标系与一坐标系重合。

(2) x_n 轴垂直于 z_n 轴并指向离开 z_n 轴的方向。

(3) y_n 轴的方向按右手定则确定。

2.构件参数的确定根据 D-H 构件坐标系表示法，构件本身的结构参数 a_{n-1} 、 α_{n-1} 和相对位置参数 d_n 、 θ_n 可由以下的方法确定:

(1) θ_n 为绕 z_n 轴(按右手定则)由 x_{n-1} 轴到 x_n 轴的关节角。

(2) d_n 为沿 z_n 轴，将 x_{n-1} 轴平移至 x_n 轴的距离。

(3) a_{n-1} 为沿 x_{n-1} 轴从 z_{n-1} 量至 z_n 轴的距离。

(4) α_{n-1} 为绕 x_{n-1} 轴(按右手定则)由 z_{n-1} 轴到 z_n 轴的偏转角。

3.变换矩阵的建立全部的连杆规定坐标系之后，就可以按照下列的顺序来建立相邻两连杆 $n-1$ 和 n 之间的相对关系:

(1)绕 x_{n-1} 轴转 α_{n-1} 角。

(2)沿 x_{n-1} 轴移动 a_{n-1} 。

(3)绕 z_n 轴转 θ_n 角。

(4)沿 z_n 轴移动 d_n 。

这种关系可由表示连杆 n 对连杆 $n-1$ 相对位置齐次变换 ${}^{n-1}T_n$ 来表征。即

$${}^{n-1}T_n = T_r(x_{n-1}, \alpha_{n-1})T_t(x_{n-1}, a_{n-1})T_r(z_n, \theta_n)T_t(z_n, d_n)$$

展开上式得

$${}^{n-1}T_n = \begin{bmatrix} \cos \theta_n & -\sin \theta_n & 0 & a_{n-1} \\ \sin \theta_n \cos \alpha_{n-1} & \cos \theta_n \cos \alpha_{n-1} & -\sin \alpha_{n-1} & -d_n \sin \alpha_{n-1} \\ \sin \theta_n \sin \alpha_{n-1} & \cos \theta_n \sin \alpha_{n-1} & \cos \alpha_{n-1} & d_n \cos \alpha_{n-1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.84)$$

由于 ${}^{n-1}T_n$ 描述第 n 个连杆相对于第 $n-1$ 连杆的位姿，对于 SCARA 教学机器人(四个自由度)，机器人的末端装置即为连杆 4 的坐标系，它与基座的关系为:

$${}^0T_4 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4$$

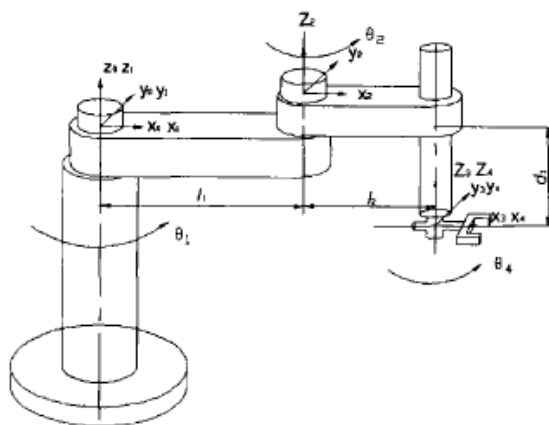


图 2.33 SCARA 机器人的 D-H 连杆坐标系的建立

Figure2.33 Construct of SCARA D-H link coordinate systems

如图 2.33 坐标系，可写出连杆 n 相对于 $n-1$ 变换矩阵 ${}^{n-1}T_n$ ：

$${}^0T_1 = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1T_2 = \begin{bmatrix} c_2 & -s_2 & 0 & l_1 \\ s_2 & c_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^2T_3 = \begin{bmatrix} 1 & 0 & 0 & l_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^3T_4 = \begin{bmatrix} c_4 & -s_4 & 0 & 0 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.85)$$

其中： $c_n = \cos \theta_n, s_n = \sin \theta_n$ 以下相同。

相应的连杆初始位置及参数列于表 2.4，表中 θ_n 、 d_n 为关节变量。

表 2.4 SCARA 机器人的杆件参数

Table2.4 Component Link Parameters of SCARE

构件	a_{n-1}	α_{n-1}	d_n	θ_n	$\cos \alpha_{n-1}$	$\sin \alpha_{n-1}$
1	0	0	0	θ_1	1	0
2	l_1	0	0	θ_2	1	0
3	l_2	0	d_3	0	1	0
4	0	0	0	θ_4	1	0

2.13.2 SCARA 机器人的正运动学分析

各连杆变换矩阵相乘，可得到 SCARA 机器人末端执行器的位姿方程(正运动学方程)为

$${}^0T_4 = {}^0T_1(\theta_1) {}^1T_2(\theta_2) {}^2T_3(d_3) {}^3T_4(\theta_4) = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.86)$$

$$= \begin{bmatrix} c_1c_2c_4 - s_1s_2s_4 - c_1s_2s_4 - s_1c_2s_4 & -c_1c_2s_4 + s_1s_2s_4 - c_1s_2c_4 - s_1c_2c_4 & 0 & c_1c_2l_2 - s_1s_2l_2 + c_1l_1 \\ s_1c_2c_4 + c_1s_2s_4 - s_1s_2s_4 + c_1c_2s_4 & -s_1c_2s_4 - c_1s_2s_4 - s_1s_2c_4 + c_1c_2c_4 & 0 & s_1c_2l_2 + c_1s_2l_2 + s_1l_1 \\ 0 & 0 & 1 & -d_3 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

式(2.86)表示了 SCARA 手臂变换矩阵 0T_4 ，它描述了末端连杆坐标系{4}相对基坐标系{0}的位姿

2.13.3 SCARA 机器人的逆运动学分析

1. 求关节变量 θ_1 为了分离变量，对方程的两边同时左乘 ${}^0T_1^{-1}(\theta_1)$ ，

得：

$${}^0T_1^{-1}(\theta_1) {}^0T_4 = {}^1T_2(\theta_2) {}^2T_3(d_3) {}^3T_4(\theta_4)$$

即：

$$\begin{bmatrix} c_1 & s_1 & 0 & 0 \\ -s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_2c_4 - s_2s_4 & -c_2s_4 - s_2c_4 & 0 & c_2l_2 + l_1 \\ s_2c_4 + c_2s_4 & -s_2s_4 + c_2c_4 & 0 & s_2l_2 \\ 0 & 0 & 1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

左右矩阵中的第一行第四个元素(1.4)，第二行第四个元素(2.4)分别相等。即：

$$\left. \begin{aligned} \cos \theta_1 \cdot p_x + \sin \theta_1 \cdot p_y &= \cos \theta_2 \cdot l_2 + l_1 \\ -\sin \theta_1 \cdot p_x + \cos \theta_1 \cdot p_y &= \sin \theta_2 \cdot l_2 \end{aligned} \right\} \quad (2.87)$$

由以上两式联立可得：

$$\theta_1 = \arctan\left(\frac{\pm\sqrt{1-A^2}}{A}\right) + \varphi \quad (2.88)$$

$$\text{式中： } A = \frac{l_1^2 - l_2^2 + p_x^2 + p_y^2}{2l_1 \cdot \sqrt{p_x^2 + p_y^2}}; \varphi = \arctan \frac{p_y}{p_x}$$

2 求关节变量 θ_2 由式(2.87)可得：

$$\theta_2 = \arctan\left[-\frac{r \sin(\theta_1 - \varphi)}{r \cos(\theta_1 - \varphi) - l_1}\right] \quad (2.89)$$

$$\text{式中： } r = \sqrt{p_x^2 + p_y^2}; \varphi = \arctan \frac{p_y}{p_x}$$

3 求关节变量 d_3 再令左右矩阵中的第三行第四个元素(3.4)相等，可得：

$$d_3 = -p_z \quad (2.90)$$

4 求关节变量 θ_4 再令左右矩阵中的第一行第一个元素、第二行第一个元素(1.1， 2.1)分别相等，即：

$$\begin{aligned} \cos \theta_1 \cdot n_x + \sin \theta_1 n_y &= \cos \theta_2 \cdot \cos \theta_4 + \sin \theta_2 \cdot \sin \theta_4 \\ -\sin \theta_1 \cdot n_x + \cos \theta_1 n_y &= \sin \theta_2 \cdot \cos \theta_4 - \cos \theta_2 \cdot \sin \theta_4 \end{aligned}$$

由上两式可求得：

$$\theta_4 = \arctan\left(\frac{-\sin\theta_1 \cdot n_x + \cos\theta_1 n_y}{\cos\theta_1 \cdot n_x + \sin\theta_1 n_y}\right) + \theta_2 \quad (2.91)$$

至此，机器人的所有运动学逆解都已求出。在逆解的求解过程中只进行了一次矩阵逆乘，从而使计算过程大为简化，从 θ_1 的表达式中可以看出它有两个解，所以 SCARA 机器人应该存在两组解。运动学分析提供了机器人运动规划和轨迹控制的理论基础。

2.14 小结

本章讨论了用矩阵表示点、向量、坐标系、及变换的方法，并利用矩阵讨论了几种特定类型机器人的正逆运动方程以及欧拉角和 RPY 姿态角，这些特定类型机器人包括直角坐标、圆柱坐标和球坐标机器人。然而，本章的主旨是学习如何表示多自由度机器人在空间的运动，以及如何用 Denavit-Hartenberg 表示法导出机器人的正逆运动学方程。这种方法可用于表示任何一种机器人的构型，而不管关节的数量和类型，以及关节和连杆的偏移和扭转。

下一章将接着讨论机器人的微分运动，实际等效于机器人的速度分析。

$$\cos\psi = f_{12}(0) \quad (3.36)$$

或

$$\sin\psi = -\sin\phi n_x + \cos\phi n_y \quad (3.37)$$

$$\cos\phi = -\sin\phi o_x + \cos\phi o_y \quad (3.38)$$

和

$$\psi = \text{atan2}(-\sin\phi n_x + \cos\phi n_y, -\sin\phi o_x + \cos\phi o_y) \quad (3.39)$$

概括起来, 当给出表示任一旋转的一个齐次变换时, 我们能够确定等效的欧拉角为

$$\phi = \text{atan2}(a_y, a_x) \quad (3.40)$$

和

$$\phi = \phi + 180^\circ \quad (3.41)$$

$$\theta = \text{atan2}(\cos\phi a_x + \sin\phi a_y, a_z) \quad (3.42)$$

$$\psi = \text{atan2}(-\sin\phi n_x + \cos\phi n_y, -\sin\phi o_x + \cos\phi o_y) \quad (3.43)$$

表示6次相乘, 3次相加和调用5 \ominus 个超越函数。

3.3 RPY变换的解

对由公式2.17~2.20确定的RPY(ϕ, θ, ψ), 我们将解方程如下

$$\text{Rot}(z, \phi)^{-1} \mathbf{T} = \text{Rot}(y, \theta) \text{Rot}(x, \psi) \quad (3.44)$$

$$\begin{bmatrix} f_{11}(n) & f_{11}(o) & f_{11}(a) & 0 \\ f_{12}(n) & f_{12}(o) & f_{12}(a) & 0 \\ f_{13}(n) & f_{13}(o) & f_{13}(a) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \sin\psi & \sin\theta \cos\psi & 0 \\ 0 & \cos\psi & -\sin\psi & 0 \\ -\sin\theta & \cos\theta \sin\psi & \cos\theta \cos\psi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.45)$$

式中

$$f_{11} = \cos\phi x + \sin\phi y \quad (3.46)$$

$$f_{12} = -\sin\phi x + \cos\phi y \quad (3.47)$$

$$f_{13} = z \quad (3.48)$$

由 $f_{12}(n)$ 元素的相等可得

\ominus 原书误作3个——校注。

$$-\sin\phi n_x + \cos\phi n_y = 0 \quad (3.49)$$

于是

$$\phi = \text{atan2}(n_y, n_x) \quad (3.50)$$

和

$$\phi = \phi + 180^\circ \quad (3.51)$$

令公式3.45两边的3,1和1,1元素相等, 我们现在得到

$$-\sin\theta = n_x \quad (3.52)$$

$$\cos\theta = \cos\phi n_x + \sin\phi n_y \quad (3.53)$$

和

$$\theta = \text{atan2}(-n_x, \cos\phi n_x + \sin\phi n_y) \quad (3.54)$$

进一步令2,3和2,2元素相等, 我们得到

$$-\sin\psi = -\sin\phi a_x + \cos\phi a_y \quad (3.55)$$

$$\cos\psi = -\sin\phi o_x + \cos\phi o_y \quad (3.56)$$

和

$$\psi = \text{atan2}(\sin\phi a_x - \cos\phi a_y, -\sin\phi o_x + \cos\phi o_y) \quad (3.57)$$

表示6次相乘, 3次相加和调用5[⊖]个超越函数。

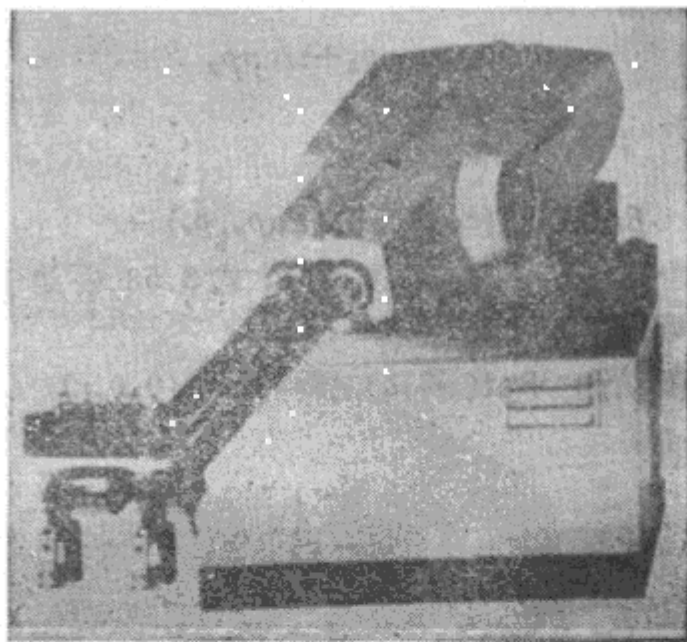


图3.2 Unimate 2100G

⊖ 原书误作3个——校注。