

机器人动力学笔记翻译

知乎 @ 疏影暗香

本小书是课程 ETH 的《Robot Dynamic Notes》的翻译

本小书翻译只做学习交流使用，英文版权归源作者所有

2022 年 7 月 23 日

序言

本课程笔记翻译是本人利用业余时间完成，从开始翻译到能够初次发布，中间断断续续过了一年，万幸最后还是坚持了下来，这才有了现在的这本初稿。本稿初步完成了笔记中的运动学和动力学部分的翻译，但大概率存在较多的错误，读者在阅读过程中如果发现相关错误可以同译者联系更改。本课程还有配套的课程作业，但译者还尚未翻译完，可能在后续的版本中添加上，敬请期待。

对本人来说，整个翻译的过程就是学习的过程。萌生将这个课程笔记翻译成中文的原因有两个：一是因为本笔记的简洁与全面，能够给入门的读者以快速地引导；二是本人对这个领域的理解不够，所以希望通过翻译整本笔记来加深对机器人动力学这个领域的了解。

虽然译者现在已经不在运动控制这个领域工作，但是双足与四足机器人依然是本人追求的一个目标，或许在不远的未来又能重新相遇。

知乎 @ 疏影暗香

深圳

2022.07.23

目录

序言	I
目录	II
第一章 导论	1
1.1 符号对照	1
1.2 操作符	3
第二章 运动学	5
2.1 简介	5
2.2 位置	5
2.2.1 位置的表示	6
2.3 线速度	7
2.3.1 线性速度的表示	7
2.4 旋转	9
2.4.1 旋转矩阵	9
2.4.2 主动旋转与被动旋转	11
2.4.3 基础旋转	12
2.4.4 复合旋转	12
2.4.5 旋转表示	12
2.5 角速度	21
2.5.1 参数化旋转的时间导数	23
2.6 变换	26
2.7 移动刚体的速度	28
2.8 刚体系统的运动学	31
2.8.1 广义坐标和关节位形	31
2.8.2 任务空间坐标	32

2.8.3	正向运动学	35
2.8.4	微分运动学和解析雅可比	36
2.8.5	几何与基本雅可比	38
2.8.6	几何矩阵与解析雅可比矩阵的关系	43
2.9	运动学控制方法	44
2.9.1	逆微分运动学	44
2.9.2	多任务逆微分运动学控制	46
2.9.3	逆运动学	50
2.9.4	轨迹控制	56
2.10	浮动基座运动学	56
2.10.1	广义速度和加速度	57
2.10.2	正向运动学	58
2.10.3	浮动基座系统的微分运动学	58
2.10.4	接触和约束	59
2.10.5	点触点-四足	60
2.10.6	扩展接触-人形	61
2.10.7	支持一致逆运动学	61
第三章	动力学	63
3.1	介绍	63
3.2	经典力学基础	64
3.2.1	牛顿粒子定律	64
3.2.2	虚位移	65
3.2.3	单个刚性体的虚位移	66
3.2.4	多体系统的虚位移	66
3.2.5	虚功原理	67
3.3	牛顿-欧拉法	67
3.3.1	单体牛顿欧拉	67
3.3.2	多体系统牛顿欧拉方程	69
3.4	拉格朗日法	70
3.4.1	介绍	70
3.4.2	动能	70
3.4.3	势能	71
3.4.4	外力	72
3.4.5	附加约束	72

3.5	投影牛顿-欧拉法	73
3.5.1	介绍	73
3.5.2	导出广义运动方程	73
3.5.3	外力和驱动	75
3.6	方法之间的总结和关系	76
3.7	浮动基座系统的动力学	76
3.7.1	接触力	77
3.7.2	约束一致动力学	78
3.7.3	接触开关和碰撞	79
3.8	关节空间动力学控制	80
3.8.1	关节阻抗调节	80
3.9	任务空间动力学控制	81
3.9.1	多任务分解	82
3.9.2	末端执行器动力学	82
3.9.3	末端执行器运动控制	83
3.9.4	操作空间控制	83
3.10	浮动基座系统逆动力学	84
3.10.1	二次问题	85
3.10.2	迭代零空间投影	85
3.10.3	序列约束优化	86
3.10.4	浮动基座系统任务空间控制 QP	86
3.11	准静态(虚拟模型)控制	87
	参考文献	91

第一章 导论

“机器人动力学”课程简要介绍了机器人系统的建模，并利用得到的模型对系统的控制做了初步尝试。本课程试图引导学习者理解不同类型机器人之间相似性，我们将涉及机械臂、腿足式机器人、轮式机器人和飞行系统，上述机器人都可以使用相同的技术来进行建模。事实上，大多数机器人动力学模型可以(足够准确地)通过作用在单个或者多个物体上的不同的力来描述。不过在不同的机器人系统中，这些力的来源不同：在自由空间中移动的机械臂所受到的力由关节驱动器提供；腿足式机器人则会在足底受到和地面的相互作用力；另一方面，在空中保持飞行状态的飞行器则是受到空气动力。

通常我们可以将机器人划分为两类，即固定基座和浮动基座机器人系统。前一类包括各种用螺栓固定在地面上的机器人，最经典的例子是工业机械臂，其主要特征是每个关节对应一个驱动器，也就意味着系统的自由度 (DOF) 等于系统中驱动器的数量。后一类包括移动机器人，即基座是移动的，与机械臂所代表的固定基座机器人不同的是，浮动基座机器人的运动通常不能通过驱动器直接控制，而只能通过作用在系统上的外力 (例如接触力或空气动力) 来控制。

本课程对应的讲义简要概述了运动学和动力学中的基本理论。由于该课程仍在完善中，因此讲义存在诸多不足，如果读者对本课程所讨论的主题感兴趣，可以参考 Handbook of Robotics^[7] (Siciliano, Khatib)¹和 Robotics - Modelling, Planning and Control^[8] (Siciliano, Sciavicco, Villani, Oriolo)²这两本教材。

1.1 符号对照

以下列表是整个讲义的符号表。几乎每本教科书都使用不同的符号对照，我们在这里遵守 IEEE 标准，并使用索引参数来确保描述的完整性。

¹<http://link.springer.com/book/10.1007%2F978-1-84628-642-1>

²<http://link.springer.com/referencework/10.1007/9783540303015>

表 1.1: 符号对照表

\mathbf{r}	向量 (小写粗体字母)
\mathbf{B}	矩阵 (大写粗体字母)
$\mathbf{e}_x^A, \mathbf{e}_y^A, \mathbf{e}_z^A$	坐标系 \mathcal{A} 的单位基向量
\mathcal{A}	坐标系 \mathcal{A} (花体字母)
A	坐标系 \mathcal{A} 的原点
$\mathbf{e}_x^I, \mathbf{e}_y^I, \mathbf{e}_z^I$	全局/惯性/世界坐标系 (固定不动)
$\mathcal{A}\mathbf{r}_{AP}$	在坐标系 \mathcal{A} 中表示的点 \mathbf{P} 相对于坐标系 \mathcal{A} 的原点 A 的向量
ϕ_{AB}	描述坐标系 \mathcal{B} 相对于坐标系 \mathcal{A} 的旋转, 属于 $SO(3)$
χ_P	表示位置的堆积参数
χ_R	表示方向的堆积参数
$\chi_{P,0}$	表示位置的最小维度的堆积参数
$\chi_{R,0}$	表示方向的最小维度的堆积参数
\mathbf{v}_P	点 P 相对于惯性系的绝对速度
\mathbf{a}_P	点 P 相对于惯性系的绝对加速度
ω_{AB}	坐标系 \mathcal{B} 相对于坐标系 \mathcal{A} 的角速度
$\Omega_B = {}^I\omega_{IB}$	坐标系 \mathcal{B} 的绝对角速度
$\Psi_B = \dot{\Omega}_B$	坐标系 \mathcal{B} 的绝对角加速度
n_q	广义坐标的总数量
n_j	关节数量
n_l	连杆数量
n_τ	驱动关节数量
n_b	基坐标数量
n_c	接触点数量
m	末端位形参数数量
m_0	操作空间坐标数量 = 末端位形最小参数数量
ϵ	加工精度
\mathbf{x}^*	星号代表期望值
\mathbf{q}	广义坐标
\mathbf{J}	雅可比矩阵
\mathbf{J}_A	分析雅可比矩阵
\mathbf{J}_0	基础雅可比矩阵
$\mathbf{N} = \mathcal{N}(\mathbf{J})$	零空间投影矩阵

1.2 操作符

在本书中，我们还经常会看到以下操作符：

$$\text{- 叉乘: } \mathbf{a} \times \mathbf{b} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \times \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = [\mathbf{a}]_{\times} \mathbf{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

$$\text{- 欧式范数: } \|\mathbf{a}\| = \sqrt{\mathbf{a}^T \mathbf{a}} = \sqrt{\mathbf{a}_1^2 + \cdots + \mathbf{a}_n^2}$$

$$\text{- 向量绝对值: } |\mathbf{a}| = (|\mathbf{a}_1| \quad \cdots \quad |\mathbf{a}_n|)$$

$$\text{- 零空间投影: } \mathcal{N}(\mathbf{J})$$

第二章 运动学

2.1 简介

运动学是用来描述点、刚体和刚体系统的运动的。它只描述了事物是如何移动的，而不讨论导致运动出现的原因。为了描述运动点的运动学，我们将引入定义在 R^3 的位置向量以及其导数。对于可旋转的刚体，我们还需要考虑其旋转，从而完全定义其位形。

在下文中，我们将首先通过描述点和单个刚体的运动来讨论运动学的基础知识，进入第 2.8 节后再介绍串联系统的运动学。

2.2 位置

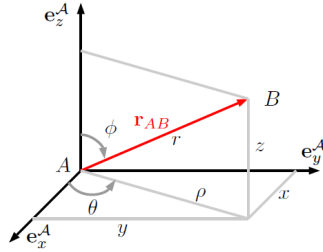


图 2.1: 使用笛卡尔、柱坐标和球坐标来表示位置

点 B 相对于点 A 的位置可以写成：

$$\mathbf{r}_{AB} \quad (2.1)$$

对于三维空间中的点，位置由向量 $\mathbf{r} \in \mathbb{R}^3$ 表示。为了数值化表示向量的分

量，有必要定义一个参考系 \mathcal{A} ，并在此坐标系中表示向量：

$$\mathcal{A}\mathbf{r}_{AB} \quad (2.2)$$

其中单位向量 $(\mathbf{e}_x^{\mathcal{A}} \mathbf{e}_y^{\mathcal{A}} \mathbf{e}_z^{\mathcal{A}})$ 是坐标系 \mathcal{A} 在 \mathbb{R}^3 空间的正交基。

2.2.1 位置的表示

三维空间中的位置需要三个参数来表示，参数化的方式有以下几种。

笛卡尔坐标

最常见的方法是使用笛卡尔坐标来表示位置，即：

$$\chi_{P_c} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad (2.3)$$

这意味着位置向量可以简单地由以下公式表示：

$$\mathcal{A}\mathbf{r} = x\mathbf{e}_x^{\mathcal{A}} + y\mathbf{e}_y^{\mathcal{A}} + z\mathbf{e}_z^{\mathcal{A}} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad (2.4)$$

柱坐标

第二种方法是使用柱坐标作为参数表示方法：

$$\chi_{P_z} = \begin{pmatrix} \rho \\ \theta \\ z \end{pmatrix} \quad (2.5)$$

这意味着位置向量为：

$$\mathcal{A}\mathbf{r} = \begin{pmatrix} \rho \cos \theta \\ \rho \sin \theta \\ z \end{pmatrix} \quad (2.6)$$

球坐标

第三种方法是使用球坐标¹作为参数表示方法：

$$\chi_{P_s} = \begin{pmatrix} r \\ \theta \\ \phi \end{pmatrix} \quad (2.7)$$

这意味着位置向量为：

$${}_{\mathcal{A}}\mathbf{r} = \begin{pmatrix} r \cos \theta \sin \phi \\ r \sin \theta \sin \phi \\ r \cos \phi \end{pmatrix} \quad (2.8)$$

注意：我们之前讨论的参数化都需要三个参数来描述三维空间中的位置，这意味着它们同时也是**最小参数化表示**方法，但旋转的情况有所不同，我们将在第 2.4.5 小节详细介绍。在大多数情况下，人们常常使用笛卡尔坐标，因为在笛卡尔坐标下向量的微积分运算最简单。

2.3 线速度

点 B 相对于点 A 的速度为：

$$\dot{\mathbf{r}}_{AB} \quad (2.9)$$

在三维空间中，速度由向量 $\dot{\mathbf{r}} \in \mathbb{R}^3$ 表示。在对速度做向量代数运算时，其运算规则与位置是相同的。

2.3.1 线性速度的表示

速度 $\dot{\mathbf{r}}$ 和当前表示形式下位置的导数 $\dot{\chi}_P$ 之间存在线性映射关系 $\mathbf{E}_P(\chi)$ ：

$$\dot{\mathbf{r}} = \mathbf{E}_P(\chi_P) \dot{\chi}_P \quad (2.10)$$

$$\dot{\chi}_P = \mathbf{E}_P^{-1}(\chi_P) \dot{\mathbf{r}} \quad (2.11)$$

¹图 2.1 对方位角 θ 和极角 ϕ 进行了展示。在数学表示中，我们通常使用 ϕ 来替代 φ ，这是为了避免与本讲义中其他用途下的 φ 产生冲突。在物理学中，我们常使用 φ 来表示方位角和 θ 来表示极角。另外在使用 Matlab 时也要特别小心，因为 φ 通常代表的是俯仰角而不是方位角。

笛卡尔坐标

如果采用笛卡尔坐标，映射关系 $\mathbf{E}_P(\boldsymbol{\chi})$ 只是简单的单位矩阵：

$$\mathbf{E}_{P_c}(\boldsymbol{\chi}_{P_c}) = \mathbf{E}_{P_c}^{-1}(\boldsymbol{\chi}_{P_c}) = \mathbb{I} \quad (2.12)$$

柱坐标

如果采用柱坐标表示，我们可以得到速度：

$$\dot{\mathbf{r}}(\boldsymbol{\chi}_{P_z}) = \begin{pmatrix} \dot{\rho} \cos \theta - \rho \dot{\theta} \sin \theta \\ \dot{\rho} \sin \theta + \rho \dot{\theta} \cos \theta \\ \dot{z} \end{pmatrix} \quad (2.13)$$

回顾下之前柱坐标系与笛卡尔坐标系之间的关系，并对其求导得到：

$$\boldsymbol{\chi}_{P_c} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \rho \cos \theta \\ \rho \sin \theta \\ z \end{pmatrix} \Rightarrow \dot{\boldsymbol{\chi}}_{P_c} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} \dot{\rho} \cos \theta - \rho \dot{\theta} \sin \theta \\ \dot{\rho} \sin \theta + \rho \dot{\theta} \cos \theta \\ \dot{z} \end{pmatrix}$$

根据上式，可以使用笛卡尔坐标系中的导数来表示柱坐标系的导数，即可得：

$$\begin{aligned} \dot{\boldsymbol{\chi}}_{P_z} &= \begin{pmatrix} \dot{\rho} \\ \dot{\theta} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} \dot{x} \cos \theta + \dot{y} \sin \theta \\ -\dot{x} \sin \theta / \rho + \dot{y} \cos \theta / \rho \\ \dot{z} \end{pmatrix} \\ &= \underbrace{\begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta / \rho & \cos \theta / \rho & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{E}_{P_z}^{-1}} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} \end{aligned} \quad (2.14)$$

由公式 2.13 可以得到线性映射关系：

$$\mathbf{E}_{P_z}(\boldsymbol{\chi}_{P_z}) = \frac{\partial \mathbf{r}(\boldsymbol{\chi}_{P_z})}{\partial \boldsymbol{\chi}_{P_z}} = \begin{bmatrix} \cos \theta & -\rho \sin \theta & 0 \\ \sin \theta & \rho \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.15)$$

球坐标

回顾下之前球坐标系与笛卡尔坐标系之间的关系，并对其求导可得：

$$\begin{aligned}\chi_{P_c} &= \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} r \cos \theta \sin \phi \\ r \sin \theta \sin \phi \\ r \cos \phi \end{pmatrix} \\ \Rightarrow \dot{\chi}_{P_c} &= \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} \dot{r} \cos \theta \sin \phi - r \dot{\theta} \sin \theta \sin \phi - r \dot{\phi} \cos \theta \cos \phi \\ \dot{r} \sin \theta \sin \phi - r \dot{\theta} \cos \theta \sin \phi - r \dot{\phi} \sin \theta \cos \phi \\ \dot{r} \cos \phi + r \dot{\phi} \sin \phi \end{pmatrix}\end{aligned}$$

对球坐标使用相同的方法会得到：

$$\mathbf{E}_{P_s} = \begin{bmatrix} \cos \theta \sin \phi & -r \sin \phi \sin \theta & r \cos \phi \cos \theta \\ \sin \phi \sin \theta & r \cos \theta \sin \phi & r \cos \phi \sin \theta \\ \cos \phi & 0 & -r \sin \phi \end{bmatrix} \quad (2.16)$$

$$\mathbf{E}_{P_s}^{-1} = \begin{bmatrix} \cos \theta \sin \phi & \sin \phi \sin \theta & \cos \phi \\ -\sin \theta / (r \sin \phi) & \cos \theta / (r \sin \phi) & 0 \\ (\cos \phi \cos \theta) / r & (\cos \phi \sin \theta) / r & -\sin \phi / r \end{bmatrix} \quad (2.17)$$

2.4 旋转

一个点的位形可以由位置完全描述，但是对于一个刚体来说，还需要引入旋转来确定其位姿。旋转在理论上用数学公式可抽象为：

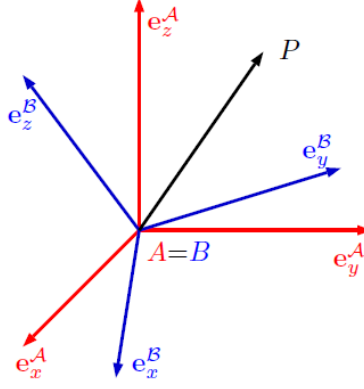
$$\phi_{AB} \in SO(3) \quad (2.18)$$

其通常用于表示固定在刚体上的坐标系 \mathcal{B} 相对于参考坐标系 \mathcal{A} 的旋转关系。理解旋转很重要，因为 ϕ_{AB} 属于 $SO(3)$ ，不存在“角位置”这样的物理参数在运算上等同于我们之前讨论的位置。相反，方向 ϕ_{AB} 可以通过好几种方式来进行参数化。为了更好地理解旋转，我们将首先通过旋转矩阵定义坐标系之间的映射关系，然后展示它们如何与不同的参数化方式产生联系。

2.4.1 旋转矩阵

图 2.2 描述了两个坐标系之间的关系，在坐标系 \mathcal{A} 中固定点 \mathbf{P} 的位置向量可以写成：

$$\mathcal{A}\mathbf{r}_{AP} = \begin{pmatrix} \mathcal{A}^r AP_x \\ \mathcal{A}^r AP_y \\ \mathcal{A}^r AP_z \end{pmatrix} \quad (2.19)$$

图 2.2: 两坐标系 \mathcal{A} 和 \mathcal{B} 之间的通用三维旋转

现在考虑一个相对于坐标系 \mathcal{A} 发生旋转的参考坐标系 \mathcal{B} ，坐标系 \mathcal{B} 的原点与坐标系 \mathcal{A} 的重合，此时坐标系 \mathcal{B} 中点的位置向量 P 是：

$$\mathcal{B}\mathbf{r}_{AP} = \begin{pmatrix} \mathcal{B}r_{AP_x} \\ \mathcal{B}r_{AP_y} \\ \mathcal{B}r_{AP_z} \end{pmatrix} \quad (2.20)$$

坐标系 \mathcal{B} 的单位向量在坐标系 \mathcal{A} 表示为 $[\mathcal{A}\mathbf{e}_x^{\mathcal{B}} \ \mathcal{A}\mathbf{e}_y^{\mathcal{B}} \ \mathcal{A}\mathbf{e}_z^{\mathcal{B}}]$ ，我们可以得到两个位置向量 $\mathcal{A}\mathbf{r}_{AP}$ 和 $\mathcal{B}\mathbf{r}_{AP}$ 之间的映射为：

$$\mathcal{A}\mathbf{r}_{AP} = \mathcal{A}\mathbf{e}_x^{\mathcal{B}} \cdot \mathcal{B}r_{AP_x} + \mathcal{A}\mathbf{e}_y^{\mathcal{B}} \cdot \mathcal{B}r_{AP_y} + \mathcal{A}\mathbf{e}_z^{\mathcal{B}} \cdot \mathcal{B}r_{AP_z} \quad (2.21)$$

公式 2.21 中所示的映射可以以紧凑的形式重写为：

$$\begin{aligned} \mathcal{A}\mathbf{r}_{AP} &= \begin{bmatrix} \mathcal{A}\mathbf{e}_x^{\mathcal{B}} & \mathcal{A}\mathbf{e}_y^{\mathcal{B}} & \mathcal{A}\mathbf{e}_z^{\mathcal{B}} \end{bmatrix} \cdot \mathcal{B}\mathbf{r}_{AP} \\ &= \mathbf{C}_{\mathcal{AB}} \cdot \mathcal{B}\mathbf{r}_{AP} \end{aligned} \quad (2.22)$$

$\mathbf{C}_{\mathcal{AB}}$ 项被称为旋转矩阵，是一个 3×3 矩阵。因为 $\mathbf{C}_{\mathcal{AB}}$ 的列向量是正交单位向量，也就是说 $\mathbf{C}_{\mathcal{AB}}$ 是正交的，意味着：

$$\mathbf{C}_{\mathcal{AB}}^T \cdot \mathbf{C}_{\mathcal{AB}} = \mathbb{I}_3 \quad (2.23)$$

公式 2.23 的结果表明 $\mathbf{C}_{\mathcal{BA}} = \mathbf{C}_{\mathcal{AB}}^{-1} = \mathbf{C}_{\mathcal{AB}}^T$ 。旋转矩阵 $\mathbf{C}_{\mathcal{AB}}$ 属于特殊正交群 $SO(3)$ ，旋转矩阵的运算不能使用 \mathbb{R}^3 空间中的经典向量代数，而是需要利用一种特殊类型的代数。

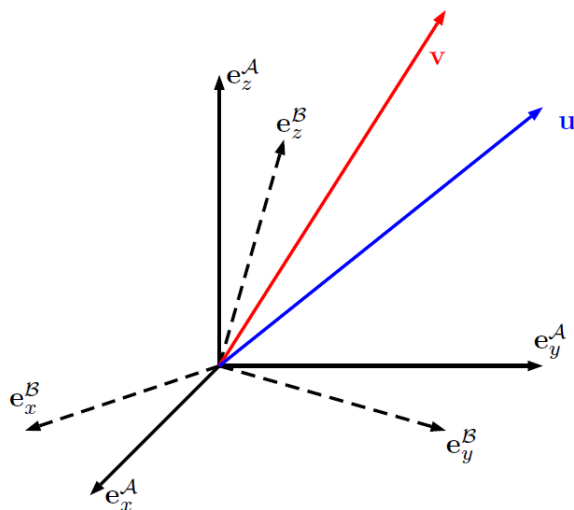


图 2.3: 被动旋转可以解释为坐标坐标系的旋转，而主动旋转可以解释为从向量 \mathbf{u} 到向量 \mathbf{v} 所对应的旋转

2.4.2 主动旋转与被动旋转

旋转可以有两种不同的解释，这引出了所谓的主动和被动旋转的定义。

被动旋转

被动旋转，也称为旋转变换，对应于公式 2.21 中的坐标系之间的变换。被动旋转 C_{AB} 将同一对象 \mathbf{u} 从坐标系 B 映射到坐标系 A ：

$${}_A\mathbf{u} = C_{AB} \cdot {}_B\mathbf{u} \quad (2.24)$$

主动旋转

主动旋转通常用 3×3 的矩阵 \mathbf{R} 表示，矩阵 \mathbf{R} 是在同一参考坐标系 A 中将向量 ${}_A\mathbf{u}$ 旋转到向量 ${}_A\mathbf{v}$ 的运算符：

$${}_A\mathbf{v} = \mathbf{R} {}_A\mathbf{u} \quad (2.25)$$

主动旋转在机器人动力学中不是很常用，因此在本课程的过程中不做过多介绍。

2.4.3 基础旋转

最简单且同时最常出现的旋转是基础旋转，即围绕坐标系的三个正交基向量 \mathbf{e}_x^A 、 \mathbf{e}_y^A 或 \mathbf{e}_z^A 中的任意一个的旋转。给定旋转角度 φ ，三个基础旋转是：

$$\mathbf{C}_{AB} = \mathbf{C}_x(\varphi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \varphi & -\sin \varphi \\ 0 & \sin \varphi & \cos \varphi \end{bmatrix} \quad (2.26)$$

$$\mathbf{C}_{AB} = \mathbf{C}_y(\varphi) = \begin{bmatrix} \cos \varphi & 0 & \sin \varphi \\ 0 & 1 & 0 \\ -\sin \varphi & 0 & \cos \varphi \end{bmatrix} \quad (2.27)$$

$$\mathbf{C}_{AB} = \mathbf{C}_z(\varphi) = \begin{bmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.28)$$

2.4.4 复合旋转

考虑三个参考坐标系 \mathcal{A} 、 \mathcal{B} 和 \mathcal{C} 。向量 \mathbf{u} 的坐标可由坐标系 \mathcal{B} 映射到坐标系 \mathcal{A} ：

$${}_{\mathcal{A}}\mathbf{u} = \mathbf{C}_{AB} \cdot {}_{\mathcal{B}}\mathbf{u} \quad (2.29)$$

我们也可以考虑向量 \mathbf{u} 的坐标在坐标系 \mathcal{B} 和坐标系 \mathcal{C} 表示之间的关系：

$${}_{\mathcal{B}}\mathbf{u} = \mathbf{C}_{BC} \cdot {}_{\mathcal{C}}\mathbf{u} \quad (2.30)$$

通过结合上面的两个方程，我们可以得到：

$$\begin{aligned} {}_{\mathcal{A}}\mathbf{u} &= \mathbf{C}_{AB} \cdot (\mathbf{C}_{BC} \cdot {}_{\mathcal{C}}\mathbf{u}) \\ &= \mathbf{C}_{AC} \cdot {}_{\mathcal{C}}\mathbf{u}. \end{aligned} \quad (2.31)$$

生成的旋转矩阵 $\mathbf{C}_{AC} = \mathbf{C}_{AB} \cdot \mathbf{C}_{BC}$ 可以解释为先旋转坐标系 \mathcal{A} 直到它与坐标系 \mathcal{B} 一致，然后旋转坐标系 \mathcal{B} 直到它与坐标系 \mathcal{C} 一致。

2.4.5 旋转表示

如前几节所述，三维空间中的一般旋转由 3×3 旋转矩阵表示，对应于 9 个参数。不过这 9 个参数不是独立的，而是受公式 2.23 中阐述的正交性条件的约束。实际上只需要三个独立的参数就能表示旋转，此时是空间旋转的**最小参数**

化表示法，例如欧拉角表示法。与此同时，我们也可以导出其他**非最小参数化表示法**，比如说轴角和单位四元数表示法。我们将简要讨论每个参数化的优缺点以及不同参数化之间的转化关系。三维旋转的更加详细的分析超出了本课程的范围，请参考文献 [2]。

欧拉角

空间旋转可以理解公式 2.26 至公式 2.28 中所定义三个基础旋转的序列组合。为了充分描述所有可能的方向，不应该围绕平行轴连续旋转两次。当围绕同一轴进行第一和第三次旋转时，此时的参数化方法称为**经典的欧拉角**。当这三个角度都不同时，我们通常指的是泰特-布赖恩表示法，万向表示法或横滚-俯仰-偏航角度表示法，这种方法经常被用于机器人学中。

注意：旋转对应的代数运算不同于我们以前学习过的经典代数运算。因此，永远不要对欧拉角、轴角或四元数进行加、减或简单的乘法操作。

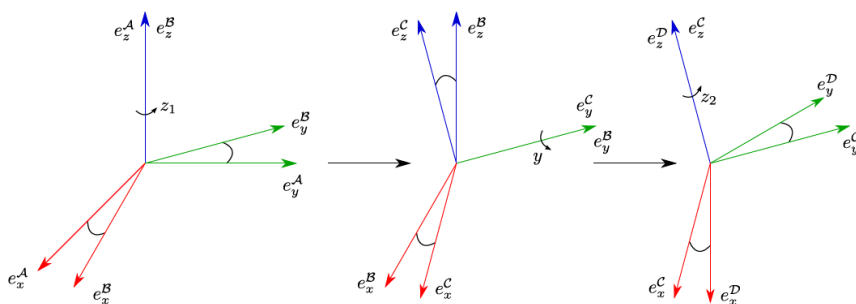


图 2.4: ZYZ 欧拉角表示绕 z 、 y 以及 z 轴的三个连续旋转的组合，对应于图中从坐标 \mathcal{A} 系到坐标系 \mathcal{D} 的旋转：(z - y' - z'')-(偏航-俯仰-偏航)

ZYZ 欧拉角 ZYZ 欧拉角是一种经典的欧拉角表示法。旋转角度可以由参数向量表示：

$$\chi_{R \text{ euler ZYZ}} = \begin{pmatrix} z_1 \\ y \\ z_2 \end{pmatrix} \quad (2.32)$$

由此产生的旋转矩阵是可以通过基础旋转的串联获得 (见图 2.4)

$$\mathbf{C}_{\mathcal{AD}} = \mathbf{C}_{\mathcal{AB}}(z_1) \mathbf{C}_{\mathcal{BC}}(y) \mathbf{C}_{\mathcal{CD}}(z_2) \Rightarrow \mathbf{A}\mathbf{r} = \mathbf{C}_{\mathcal{AD}}\mathbf{D}\mathbf{r}$$

$$\begin{aligned}
&= \begin{bmatrix} \cos z_1 & -\sin z_1 & 0 \\ \sin z_1 & \cos z_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos y & 0 & \sin y \\ 0 & 1 & 0 \\ -\sin y & 0 & \cos y \end{bmatrix} \begin{bmatrix} \cos z_2 & -\sin z_2 & 0 \\ \sin z_2 & \cos z_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} c_y c_{z_1} c_{z_2} - s_{z_1} s_{z_2} & -c_{z_2} s_{z_1} - c_y c_{z_1} s_{z_2} & c_{z_1} s_y \\ c_{z_1} s_{z_2} + c_y c_{z_2} s_{z_1} & c_{z_1} c_{z_2} - c_y s_{z_1} s_{z_2} & s_y s_{z_1} \\ -c_{z_2} s_y & s_y s_{z_2} & c_y \end{bmatrix} \quad (2.33)
\end{aligned}$$

分析公式 2.33 可以找到逆问题的解，即给定旋转矩阵：

$$\mathbf{C}_{\mathcal{A}\mathcal{D}} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \quad (2.34)$$

对应的 ZYZ 欧拉角为

$$\chi_{R \text{ euler } ZYZ} = \begin{pmatrix} z_1 \\ y \\ z_2 \end{pmatrix} = \begin{pmatrix} \text{atan2}(c_{23} \ c_{13}) \\ \text{atan2}\left(\sqrt{c_{13}^2 + c_{23}^2} \ c_{33}\right) \\ \text{atan2}(c_{32} \ -c_{31}) \end{pmatrix} \quad (2.35)$$

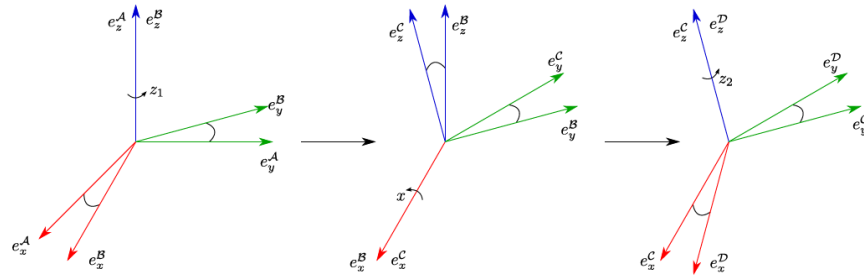


图 2.5: ZYZ 欧拉角表示沿着 z 、 x 以及 z 轴的三个连续旋转。即从上图中的 \mathcal{A} 坐标系变换到 \mathcal{D} 坐标系：(($z-x'-z''$))- (偏航-横滚-偏航)

ZXZ 欧拉角 ZXZ 欧拉角也是经典的欧拉角的一种。旋转角度可以由参数向量表示：

$$\chi_{R \text{ euler } ZXZ} = \begin{pmatrix} z_1 \\ x \\ z_2 \end{pmatrix} \quad (2.36)$$

由此产生的旋转矩阵是可以通过基础旋转 (见图 2.5) 的串联获得的:

$$\begin{aligned}
 \mathbf{C}_{AD} &= \mathbf{C}_{AB}(z_1) \mathbf{C}_{BC}(x) \mathbf{C}_{CD}(z_2) \Rightarrow \mathcal{A}\mathbf{r} = \mathbf{C}_{AD}\mathcal{D}\mathbf{r} \\
 &= \begin{bmatrix} \cos z_1 & -\sin z_1 & 0 \\ \sin z_1 & \cos z_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos x & -\sin x \\ 0 & \sin x & \cos x \end{bmatrix} \begin{bmatrix} \cos z_2 & -\sin z_2 & 0 \\ \sin z_2 & \cos z_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} c_{z_1}c_{z_2} - c_x s_{z_1}s_{z_2} & -c_{z_1}s_{z_2} - c_x c_{z_2}s_{z_1} & s_x s_{z_1} \\ c_{z_2}s_{z_1} + c_x c_{z_1}s_{z_2} & c_x c_{z_1}c_{z_2} - s_{z_1}s_{z_2} & -c_{z_1}s_x \\ s_x s_{z_2} & c_{z_2}s_x & c_x \end{bmatrix}. \quad (2.37)
 \end{aligned}$$

分析公式 2.37 可以找到逆问题的解, 即给定一个旋转矩阵:

$$\mathbf{C}_{AD} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \quad (2.38)$$

ZXZ 欧拉角由下公式可以得到:

$$\chi_{R \text{ euler } ZXZ} = \begin{pmatrix} z_1 \\ x \\ z_2 \end{pmatrix} = \begin{pmatrix} \text{atan2}(c_{13} - c_{23}) \\ \text{atan2}(\sqrt{c_{13}^2 + c_{23}^2} c_{33}) \\ \text{atan2}(c_{31} c_{32}) \end{pmatrix} \quad (2.39)$$

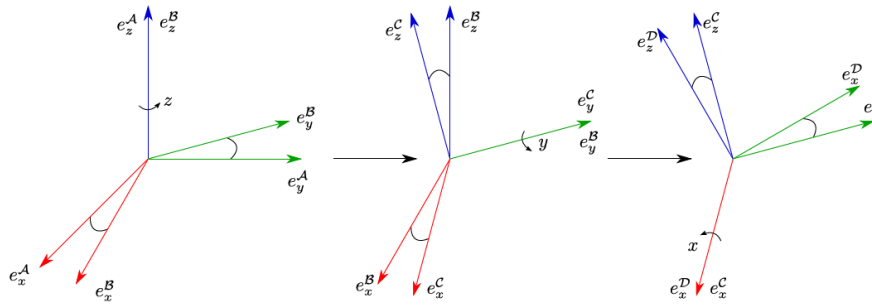


图 2.6: ZYX 欧拉角为围绕 z 、 y 和 x 轴的三个连续旋转。即从上图中的 \mathcal{A} 坐标系变换到 \mathcal{D} 坐标系: (z - y' - x'')-(偏航-俯仰-横滚)

ZYX 欧拉角 ZYX 欧拉角, 也被称为泰特布赖恩角, 通常用于描述飞行

器的姿态，又被称为偏航-俯仰-滚动。旋转角度可以由参数向量表示：

$$\chi_{R \text{ euler } ZYX} = \begin{pmatrix} z \\ y \\ x \end{pmatrix} \quad (2.40)$$

最终的旋转矩阵由基础旋转串联（见图 2.6）得到的：

$$\begin{aligned} \mathbf{C}_{AD} &= \mathbf{C}_{AB}(z)\mathbf{C}_{BC}(y)\mathbf{C}_{CD}(x) \Rightarrow \mathbf{A}\mathbf{r} = \mathbf{C}_{AD}\mathbf{D}\mathbf{r} \\ &= \begin{bmatrix} \cos z & -\sin z & 0 \\ \sin z & \cos z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos y & 0 & \sin y \\ 0 & 1 & 0 \\ -\sin y & 0 & \cos y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos x & -\sin x \\ 0 & \sin x & \cos x \end{bmatrix} \\ &= \begin{bmatrix} c_y c_z & c_z s_x s_y - c_x s_z & s_x s_z + c_x c_z s_y \\ c_y s_z & c_x c_z + s_x s_y s_z & c_x s_y s_z - c_z s_x \\ -s_y & c_y s_x & c_x c_y \end{bmatrix}. \end{aligned} \quad (2.41)$$

给定公式 2.38 中的旋转矩阵，逆解为

$$\chi_{R \text{ euler } ZYX} = \begin{pmatrix} z \\ y \\ x \end{pmatrix} = \begin{pmatrix} \text{atan2}(c_{21} \ c_{11}) \\ \text{atan2}\left(-c_{31} \ \sqrt{c_{32}^2 + c_{33}^2}\right) \\ \text{atan2}(c_{32} \ c_{33}) \end{pmatrix} \quad (2.42)$$

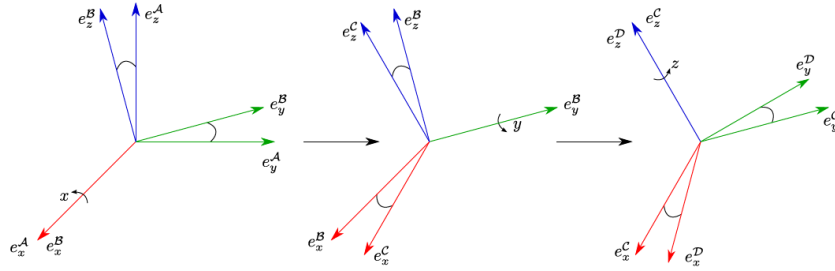


图 2.7: XYZ 欧拉角表示的是沿 x、y 以及 z 轴的三个连续旋转，即从上图中的 A 坐标系变换到 D 坐标系：(x-y'-z'')-(横滚-俯仰-偏航)

XYZ 欧拉角 XYZ 欧拉角也被称为万向角。旋转角度可以在参数向量中得到：

$$\chi_{R \text{ euler } XYZ} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad (2.43)$$

由此产生的旋转矩阵是通过基础旋转 (见图 2.7) 的串联获得的:

$$\begin{aligned}
 \mathbf{C}_{AD} &= \mathbf{C}_{AB}(x)\mathbf{C}_{BC}(y)\mathbf{C}_{CD}(z) \Rightarrow \mathbf{A}\mathbf{r} = \mathbf{C}_{AD}\mathbf{D}\mathbf{r} \\
 &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos x & -\sin x \\ 0 & \sin x & \cos x \end{bmatrix} \begin{bmatrix} \cos y & 0 & \sin y \\ 0 & 1 & 0 \\ -\sin y & 0 & \cos y \end{bmatrix} \begin{bmatrix} \cos z & -\sin z & 0 \\ \sin z & \cos z & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} c_y c_z & -c_y s_z & s_y \\ c_x s_z + c_z s_x s_y & c_x c_z - s_x s_y s_z & -c_y s_x \\ s_x s_z - c_x c_z s_y & c_z s_x + c_x s_y s_z & c_x c_y \end{bmatrix} \quad (2.44)
 \end{aligned}$$

给定公式 2.38 中的旋转矩阵, 逆解为:

$$\chi_{R \text{ euler } XYZ} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \text{atan2}(-c_{23} \ c_{33}) \\ \text{atan2}(c_{13} \ \sqrt{c_{11}^2 + c_{12}^2}) \\ \text{atan2}(-c_{12} \ c_{11}) \end{pmatrix} \quad (2.45)$$

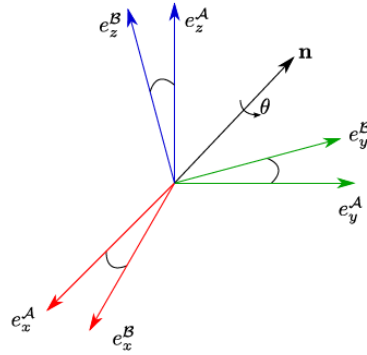


图 2.8: 轴角表示将两个坐标系的旋转定义为绕轴 \mathbf{n} 旋转角度 θ 的旋转

轴角

轴角是由角度 θ 和旋转轴 \mathbf{n} 定义的非最小参数化表示旋转的方法。向量 $\mathbf{n} \in \mathbb{R}^3$ 定义旋转的方向, 而标量 $\theta \in \mathbb{R}$ 定义旋转角度:

$$\chi_{R \text{ Angle Axis}} = \begin{pmatrix} \theta \\ \mathbf{n} \end{pmatrix} \quad (2.46)$$

这个参数化形式有四个参数和一个长度约束 $\|\mathbf{n}\| = 1$ 。可以将这两个量结合起来以获得旋转向量或欧拉向量：

$$\varphi = \theta \cdot \mathbf{n} \in \mathbb{R}^3 \quad (2.47)$$

值得注意的是, 尽管 φ 属于 \mathbb{R}^3 , 其求和运算通常是不满足交换律的, 即 $\varphi_1 + \varphi_2 \neq \varphi_2 + \varphi_1$ 。轴角参数 φ_{AB} 对应的旋转矩阵为:

$$\mathbf{C}_{AB} = \begin{bmatrix} n_x^2(1 - c_\theta) + c_\theta & n_x n_y(1 - c_\theta) - n_z s_\theta & n_x n_z(1 - c_\theta) + n_y s_\theta \\ n_x n_y(1 - c_\theta) + n_z s_\theta & n_y^2(1 - c_\theta) + c_\theta & n_y n_z(1 - c_\theta) - n_x s_\theta \\ n_x n_z(1 - c_\theta) - n_y s_\theta & n_y n_z(1 - c_\theta) + n_x s_\theta & n_z^2(1 - c_\theta) + c_\theta \end{bmatrix} \quad (2.48)$$

给定公式 2.38 中的旋转矩阵, 对应的轴角参数为:

$$\theta = \cos^{-1} \left(\frac{c_{11} + c_{22} + c_{33} - 1}{2} \right) \quad (2.49)$$

$$\mathbf{n} = \frac{1}{2 \sin(\theta)} \begin{pmatrix} c_{32} - c_{23} \\ c_{13} - c_{31} \\ c_{21} - c_{12} \end{pmatrix} \quad (2.50)$$

从公式 2.50 可以看出, 该表示在 $\theta = 0$ 和 $\theta = \pi$ 时, 会因为 $\sin(\theta) = 0$ 而导致旋转向量未定义。对于 $\theta = 0$ 来说, 它可以表示任何方向; 而 $\theta = \pi$ 则意味着它可以代表两个相反的方向。

单位四元数

单位四元数 (也称为欧拉参数) 提供了旋转的非最小表示方法, 没有轴角表示的缺点。考虑旋转向量 $\varphi \in \mathbb{R}^3$, 单位四元数 ξ 定义为:

$$\chi_{R \text{ quat}} = \xi = \begin{pmatrix} \xi_0 \\ \check{\xi} \end{pmatrix} \in \mathbb{H} \quad (2.51)$$

其中 ξ_0 和 $\check{\xi}$ 分别为:

$$\begin{aligned} \xi_0 &= \cos \left(\frac{\|\varphi\|}{2} \right) = \cos \left(\frac{\theta}{2} \right) \\ \check{\xi} &= \sin \left(\frac{\|\varphi\|}{2} \right) \frac{\varphi}{\|\varphi\|} = \sin \left(\frac{\theta}{2} \right) \mathbf{n} = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix}. \end{aligned} \quad (2.52)$$

第一个参数 ξ_0 被称为四元数的实部, $\check{\xi}$ 为虚部。单位四元数需要满足约束:

$$\xi_0^2 + \xi_1^2 + \xi_2^2 + \xi_3^2 = 1 \quad (2.53)$$

与轴角类似，由四元数计算得到旋转矩阵是：

$$\begin{aligned} \mathbf{C}_{AD} &= \mathbb{I}_{3 \times 3} + 2\xi_0[\check{\xi}]_{\times} + 2[\check{\xi}]_{\times}^2 = (2\xi_0^2 - 1)\mathbb{I}_{3 \times 3} + 2\xi_0[\check{\xi}]_{\times} + 2\check{\xi}\check{\xi}^T \\ &= \begin{bmatrix} \xi_0^2 + \xi_1^2 - \xi_2^2 - \xi_3^2 & 2\xi_1\xi_2 - 2\xi_0\xi_3 & 2\xi_0\xi_2 + 2\xi_1\xi_3 \\ 2\xi_0\xi_3 + 2\xi_1\xi_2 & \xi_0^2 - \xi_1^2 + \xi_2^2 - \xi_3^2 & 2\xi_2\xi_3 - 2\xi_0\xi_1 \\ 2\xi_1\xi_3 - 2\xi_0\xi_2 & 2\xi_0\xi_1 + 2\xi_2\xi_3 & \xi_0^2 - \xi_1^2 - \xi_2^2 + \xi_3^2 \end{bmatrix} \end{aligned} \quad (2.54)$$

给定公式 2.38 中的旋转矩阵，相应的四元数是：

$$\chi_{R \text{ quat}} = \xi_{AD} = \frac{1}{2} \begin{pmatrix} \sqrt{c_{11} + c_{22} + c_{33} + 1} \\ \text{sgn}(c_{32} - c_{23}) \sqrt{c_{11} - c_{22} - c_{33} + 1} \\ \text{sgn}(c_{13} - c_{31}) \sqrt{c_{22} - c_{33} - c_{11} + 1} \\ \text{sgn}(c_{21} - c_{12}) \sqrt{c_{33} - c_{11} - c_{22} + 1} \end{pmatrix} \quad (2.55)$$

在我们使用四元数时，我们可以利用对应的特殊的代数运算，从而让我们直接使用四元数参数化（而不仅仅是旋转矩阵）来进行运算，感兴趣的读者可以参考文献 [9]，它提供了非常完整紧凑且易于理解的四元数介绍。简而言之，以下规则很重要，根据 $\mathbf{C}^{-1} = \mathbf{C}^T$ ，得到逆参数形式 ξ^{-1} 为：

$$\xi = \begin{pmatrix} \xi \\ \check{\xi} \end{pmatrix} \xrightarrow{\text{inverse}} \xi^{-1} = \xi^T = \begin{pmatrix} \xi \\ -\check{\xi} \end{pmatrix} \quad (2.56)$$

如果 ξ_{AB} 和 ξ_{BC} 分别表示对应于 \mathbf{C}_{AB} 和 \mathbf{C}_{BC} 的四元数，根据 $\mathbf{C}_{AC} = \mathbf{C}_{AB}\mathbf{C}_{BC}$ ，四元数的乘法运算法则为：

$$\xi_{AB} \otimes \xi_{BC} = \begin{pmatrix} \xi_{0AB} \cdot \xi_{0BC} - \check{\xi}_{AB}^T \cdot \check{\xi}_{BC} \\ \xi_{0AB} \cdot \check{\xi}_{BC} + \xi_{0BC} \cdot \check{\xi}_{AB} + [\check{\xi}_{AB}]_{\times} \cdot \check{\xi}_{BC} \end{pmatrix} \quad (2.57)$$

$$= \underbrace{\begin{bmatrix} \xi_0 & -\check{\xi}^T \\ \check{\xi} & \xi_0\mathbb{I} + [\check{\xi}]_{\times} \end{bmatrix}}_{\mathbf{M}_l(\xi_{AB})}_{AB} \begin{pmatrix} \xi_0 \\ \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix}_{BC} \quad (2.58)$$

$$= \begin{bmatrix} \xi_0 & -\xi_1 & -\xi_2 & -\xi_3 \\ \xi_1 & \xi_0 & -\xi_3 & \xi_2 \\ \xi_2 & \xi_3 & \xi_0 & -\xi_1 \\ \xi_3 & -\xi_2 & \xi_1 & \xi_0 \end{bmatrix}_{AB} \begin{pmatrix} \xi_0 \\ \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix}_{BC} \quad (2.59)$$

其中 $\mathbf{M}_l(\xi)$ 表示四元数的左矩阵。同样的四元数乘法也可以用右四元数来构造：

$$\xi_{AB} \otimes \xi_{BC} = \underbrace{\begin{bmatrix} \xi_0 & -\check{\xi}^T \\ \check{\xi} & \xi_0 \mathbb{I} - [\check{\xi}]_{\times} \end{bmatrix}}_{\mathbf{M}_r(\xi_{BC})} \begin{pmatrix} \xi_0 \\ \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix}_{AB} \quad (2.60)$$

$$= \begin{bmatrix} \xi_0 & -\xi_1 & -\xi_2 & -\xi_3 \\ \xi_1 & \xi_0 & \xi_3 & -\xi_2 \\ \xi_2 & -\xi_3 & \xi_0 & \xi_1 \\ \xi_3 & \xi_2 & -\xi_1 & \xi_0 \end{bmatrix}_{BC} \begin{pmatrix} \xi_0 \\ \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix}_{AB} \quad (2.61)$$

使用对应于四元数的特殊代数可以用于直接旋转向量。在坐标系 B 中的向量 ${}_B \mathbf{r}$ 的坐标对应的纯（虚）四元数为：

$$\mathbf{p}({}_B \mathbf{r}) = \begin{pmatrix} 0 \\ {}_B \mathbf{r} \end{pmatrix} \quad (2.62)$$

单位四元数 ξ_{AB} 代表坐标系 A 相对于坐标系 B 的旋转，可以证明 [9]：

$$\mathbf{p}({}_A \mathbf{r}) = \xi_{AB} \otimes \mathbf{p}({}_B \mathbf{r}) \otimes \xi_{AB}^T \quad (2.63)$$

$$= \mathbf{M}_l(\xi_{AB}) \mathbf{M}_r(\xi_{AB}^T) \mathbf{p}({}_B \mathbf{r}) \quad (2.64)$$

【实例 2.4.1】四元数乘法表示旋转

已知坐标系 B 由坐标系 A 绕 x 轴旋转 $\theta = \pi/3$ 得到，在坐标系 A 中有一个向量 ${}_A \mathbf{r} = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}$ ，其在坐标系 B 中表示的向量是什么？

从坐标系 B 到坐标系 A 旋转对应的四元数是

$$\xi_{AB} = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) \mathbf{n} \end{pmatrix} = \frac{1}{2} \begin{pmatrix} \sqrt{3} \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad (2.65)$$

因此，对应于逆旋转的四元数是：

$$\xi_{BA} = \xi_{AB}^T = \frac{1}{2} \begin{pmatrix} \sqrt{3} \\ -1 \\ 0 \\ 0 \end{pmatrix} \quad (2.66)$$

为了在坐标系 B 表示向量 \mathbf{r} ，我们可以应用以下计算：

$$\mathbf{p}(B\mathbf{r}) = \xi_{BA} \otimes \mathbf{p}(A\mathbf{r}) \otimes \xi_{BA}^T \quad (2.67)$$

$$= \mathbf{M}_l(\xi_{BA}) \mathbf{M}_r(\xi_{BA}^T) \mathbf{p}(B\mathbf{r}) \quad (2.68)$$

$$\begin{aligned} &= \frac{1}{2} \begin{bmatrix} \sqrt{3} & 1 & 0 & 0 \\ -1 & \sqrt{3} & 0 & 0 \\ 0 & 0 & \sqrt{3} & 1 \\ 0 & 0 & -1 & \sqrt{3} \end{bmatrix} \frac{1}{2} \begin{bmatrix} \sqrt{3} & -1 & 0 & 0 \\ 1 & \sqrt{3} & 0 & 0 \\ 0 & 0 & \sqrt{3} & 1 \\ 0 & 0 & -1 & \sqrt{3} \end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ 0 \\ \frac{1}{2} \\ -\frac{\sqrt{3}}{2} \end{pmatrix} \quad (2.69) \end{aligned}$$

2.5 角速度

考虑一个相对于固定坐标系 A 运动的坐标系 B ，使用角速度 ${}_A\omega_{AB}$ 描述坐标系 B 相对于坐标系 A 的旋转运动，其由极限定义：

$${}_A\omega_{AB} = \lim_{\epsilon \rightarrow 0} \frac{{}_A\varphi_{B(t)B(t+\epsilon)}}{\epsilon} \quad (2.70)$$

如上一节所述，旋转向量 φ 不是一个传统意义上的向量。不过当 $\epsilon \rightarrow 0$ ，角速度的定义变成传统向量 ${}_A\varphi_{B(t)B(t+\epsilon)}$ 和一个标量的比值。因此，可以根据向量求和规则来对角速度求和。因此，坐标系 A 相对于坐标系 B 的相对旋转是：

$$\omega_{AB} = -\omega_{BA} \quad (2.71)$$

可以证明角速度向量 ${}_A\omega_{AB}$ 和随时间变化的旋转矩阵 $C_{AB}(t)$ 之间的关系为：

$$[{}_A\omega_{AB}]_{\times} = \dot{C}_{AB} \cdot C_{AB}^T \quad (2.72)$$

其中 $[\mathcal{A}\boldsymbol{\omega}_{AB}]_{\times}$ 是 $\mathcal{A}\boldsymbol{\omega}_{AB}$ 对应的反对称矩阵:

$$[\mathcal{A}\boldsymbol{\omega}_{AB}]_{\times} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad \mathcal{A}\boldsymbol{\omega}_{AB} = \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} \quad (2.73)$$

角速度可以像位置向量一样通过旋转矩阵进行转换:

$$\mathcal{B}\boldsymbol{\omega}_{AB} = \mathbf{C}_{BA} \cdot \mathcal{A}\boldsymbol{\omega}_{AB} \quad (2.74)$$

相应的叉乘矩阵 $[\omega]_{\times}$ 的变换关系为:

$$[\mathcal{B}\boldsymbol{\omega}_{AB}]_{\times} = \mathbf{C}_{BA} \cdot [\mathcal{A}\boldsymbol{\omega}_{AB}]_{\times} \cdot \mathbf{C}_{AB} \quad (2.75)$$

连续坐标系的角速度可以由以下公式计算得到:

$$\mathcal{D}\boldsymbol{\omega}_{AC} = \mathcal{D}\boldsymbol{\omega}_{AB} + \mathcal{D}\boldsymbol{\omega}_{BC} \quad (2.76)$$

与之前使用的向量加法一样, 所有向量都在同一参考系统 \mathcal{D} 中表达是非常重要的。

【实例 2.5.1】从旋转矩阵得到角速度

可以使用绕 \mathbf{e}_x^A 旋转 $\alpha(t)$ 的基础旋转来表示坐标系 \mathcal{B} 相对于坐标系 \mathcal{A} 的角速度, 相应的旋转矩阵:

$$\mathbf{C}_{AB}(t) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha(t)) & -\sin(\alpha(t)) \\ 0 & \sin(\alpha(t)) & \cos(\alpha(t)) \end{bmatrix} \quad (2.77)$$

$$[\mathcal{A}\boldsymbol{\omega}_{AB}]_{\times} = \dot{\mathbf{C}}_{AB} \mathbf{C}_{AB}^T \quad (2.78)$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\dot{\alpha} \sin \alpha & -\dot{\alpha} \cos \alpha \\ 0 & \dot{\alpha} \cos \alpha & -\dot{\alpha} \sin \alpha \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha \\ 0 & -\sin \alpha & \cos \alpha \end{bmatrix} \quad (2.79)$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -\dot{\alpha} \\ 0 & \dot{\alpha} & 0 \end{bmatrix} \quad (2.80)$$

反对称矩阵转化为角速度：

$${}_{\mathcal{A}}\boldsymbol{\omega}_{\mathcal{AB}} = \begin{pmatrix} \dot{\alpha} \\ 0 \\ 0 \end{pmatrix} \quad (2.81)$$

2.5.1 参数化旋转的时间导数

如第 2.4.5 节中介绍的，存在不同的旋转参数化方法。与线速度类似，它们的导数可以映射到角速度：

$${}_{\mathcal{A}}\boldsymbol{\omega}_{\mathcal{AB}} = \mathbf{E}_R(\chi_R) \cdot \dot{\chi}_R \quad (2.82)$$

在下文中，将导出并讨论这些映射。

欧拉角 ZYX 的时间导数 \Leftrightarrow 角速度

给定一组 ZYX 欧拉角 $\chi_{R \text{ euler ZYX}} = \begin{bmatrix} z & y & x \end{bmatrix}^T$ 以及它们的导数 $\dot{\chi}_{R \text{ euler ZYX}} = \begin{bmatrix} \dot{z} & \dot{y} & \dot{x} \end{bmatrix}^T$ ，我们希望找到一个映射 $\mathbf{E}_R \text{ euler ZYX} = \mathbf{E}_R(\chi_{R \text{ euler ZYX}}) \in \mathbb{R}^{3 \times 3}$ ，其能够将 $\dot{\chi}$ 映射到 ${}_{\mathcal{A}}\boldsymbol{\omega}_{\mathcal{AB}}$ 。 $\mathbf{E}(\chi_{R \text{ euler ZYX}})$ 的列向量是在固定坐标系 \mathcal{A} 中表示的施加了角速度的单位向量的分量。这些是通过选择由欧拉角参数化指定的连续基础旋转建立的旋转矩阵的列来获得的。

从参考坐标系 \mathcal{A} 开始，第一次旋转将是绕 ${}_{\mathcal{A}}\mathbf{e}_z^{\mathcal{A}}$ 的基础旋转，由以下给定：

$${}_{\mathcal{A}}\mathbf{e}_z^{\mathcal{A}} = \mathbb{I}_{3 \times 3} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (2.83)$$

在绕 ${}_{\mathcal{A}}\mathbf{e}_z^{\mathcal{A}}$ 的基础旋转之后， y 轴 ${}_{\mathcal{A}}\mathbf{e}_y^{\mathcal{A}'}$ 将由以下公式表示：

$${}_{\mathcal{A}}\mathbf{e}_y^{\mathcal{A}'} = \mathbf{C}_{\mathcal{AA}'}(z) \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \cos(z) & -\sin(z) & 0 \\ \sin(z) & \cos(z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} -\sin(z) \\ \cos(z) \\ 0 \end{bmatrix} \quad (2.84)$$

在完成绕 ${}_{\mathcal{A}}\mathbf{e}_y^{\mathcal{A}'}$ 的基础旋转之后， x 轴 ${}_{\mathcal{A}}\mathbf{e}_x^{\mathcal{A}''}$ 将由以下公式表示：

$${}_{\mathcal{A}}\mathbf{e}_x^{\mathcal{A}''} = \mathbf{C}_{\mathcal{AA}'}(z) \cdot \mathbf{C}_{\mathcal{A}'\mathcal{A}''}(y) \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{aligned}
&= \begin{bmatrix} \cos(z) & -\sin(z) & 0 \\ \sin(z) & \cos(z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(y) & 0 & \sin(y) \\ 0 & 1 & 0 \\ -\sin(y) & 0 & \cos(y) \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} \cos(y) \cos(z) \\ \cos(y) \sin(z) \\ -\sin(y) \end{bmatrix} \tag{2.85}
\end{aligned}$$

最后映射 $\mathbf{E}(\chi_R)$ 将按照如下公式计算：

$$\mathbf{E}_{R \text{ euler } ZYX} = \begin{bmatrix} \mathcal{A}\mathbf{e}_z^{\mathcal{A}} & \mathcal{A}\mathbf{e}_y^{\mathcal{A}'} & \mathcal{A}\mathbf{e}_x^{\mathcal{A}''} \end{bmatrix} = \begin{bmatrix} 0 & -\sin(z) & \cos(y) \cos(z) \\ 0 & \cos(z) & \cos(y) \sin(z) \\ 1 & 0 & -\sin(y) \end{bmatrix} \tag{2.86}$$

很容易发现 $\det(\mathbf{E}_{R \text{ euler } ZYX}) = -\cos(y)$ 。在 $y = \pi/2 + k\pi \ \forall k \in \mathbb{Z}$ 时，映射矩阵是奇异矩阵。这意味着，尽管我们总是可以使用欧拉角相对于时间的导数来描述角速度，但其逆并不总是可以求解的。逆映射由以下给出：

$$\mathbf{E}_{R \text{ euler } ZYX}^{-1} = \begin{bmatrix} \frac{\cos(z) \sin(y)}{\cos(y)} & \frac{\sin(y) \sin(z)}{\cos(y)} & 1 \\ -\sin(z) & \cos(z) & 0 \\ \frac{\cos(z)}{\cos(y)} & \frac{\sin(z)}{\cos(y)} & 0 \end{bmatrix} \tag{2.87}$$

欧拉角 XYZ 的时间导数 \Leftrightarrow 角速度

与之前的推导类似，XYZ 欧拉角的投影矩阵及其逆

$$\mathbf{E}_{R \text{ euler } XYZ} = \begin{bmatrix} 1 & 0 & \sin(y) \\ 0 & \cos(x) & -\cos(y) \sin(x) \\ 0 & \sin(x) & \cos(x) \cos(y) \end{bmatrix} \tag{2.88}$$

$$\mathbf{E}_{R \text{ euler } XYZ}^{-1} = \begin{bmatrix} 1 & \frac{\sin(x) \sin(y)}{\cos(y)} & \frac{-\cos(x) \sin(y)}{\cos(y)} \\ 0 & \cos(x) & \sin(x) \\ 0 & \frac{-\sin(x)}{\cos(y)} & \frac{\cos(x)}{\cos(y)} \end{bmatrix} \tag{2.89}$$

欧拉角 ZYZ 的时间导数 \Leftrightarrow 角速度

ZYZ 欧拉角的投影矩阵及其逆

$$\mathbf{E}_{R \text{ euler } ZYZ} = \begin{bmatrix} 0 & -\sin(z_1) & \cos(z_1) \sin(y) \\ 0 & \cos(z_1) & \sin(z_1) \sin(y) \\ 1 & 0 & \cos(y) \end{bmatrix} \tag{2.90}$$

$$\mathbf{E}_{R \text{ eulerZYZ}}^{-1} = \begin{bmatrix} \frac{-\cos(y)\cos(z_1)}{\sin(y)} & \frac{-\cos(y)\sin(z_1)}{\sin(y)} & 1 \\ -\sin(z_1) & \cos(z_1) & 0 \\ \frac{\cos(z_1)}{\sin(y)} & \frac{\sin(z_1)}{\sin(y)} & 0 \end{bmatrix} \quad (2.91)$$

欧拉角 ZXX 的时间导数 \Leftrightarrow 角速度

ZXX 欧拉角的投影矩阵及其逆:

$$\mathbf{E}_{R \text{ euler ZXX}} = \begin{bmatrix} 0 & \cos(z_1) & \sin(z_1)\sin(x) \\ 0 & \sin(z_1) & -\cos(z_1)\sin(x) \\ 1 & 0 & \cos(x) \end{bmatrix} \quad (2.92)$$

$$\mathbf{E}_{R \text{ euler ZXX}}^{-1} = \begin{bmatrix} \frac{-\cos(x)\sin(z_1)}{\sin(x)} & \frac{\cos(x)\cos(z_1)}{\sin(x)} & 1 \\ \cos(z_1) & \sin(z_1) & 0 \\ \frac{\sin(z_1)}{\sin(x)} & \frac{-\cos(z_1)}{\sin(x)} & 0 \end{bmatrix}. \quad (2.93)$$

旋转四元数的时间导数 \Leftrightarrow 角速度

对于四元数, 可以证明以下关系成立:

$${}^I\boldsymbol{\omega}_{IB} = 2\mathbf{H}(\boldsymbol{\xi}_{IB})\dot{\boldsymbol{\xi}}_{IB} = \mathbf{E}_{R \text{ quat}}\dot{\boldsymbol{\chi}}_{R \text{ quat}} \quad (2.94)$$

$$\dot{\boldsymbol{\xi}}_{IB} = \frac{1}{2}\mathbf{H}(\boldsymbol{\xi}_{IB})^T\boldsymbol{\omega}_{IB} = \mathbf{E}_{R \text{ quat}}^{-1}\dot{\boldsymbol{\chi}}_{R \text{ quat}} \quad (2.95)$$

其中:

$$\mathbf{H}(\boldsymbol{\xi}) = \begin{bmatrix} -\check{\boldsymbol{\xi}} & [\check{\boldsymbol{\xi}}]_{\times} + \xi_0\mathbb{I}_{3\times 3} \end{bmatrix} \in \mathbb{R}^{3\times 4} \quad (2.96)$$

$$= \begin{bmatrix} -\xi_1 & \xi_0 & -\xi_3 & \xi_2 \\ -\xi_2 & \xi_3 & \xi_0 & -\xi_1 \\ -\xi_3 & -\xi_2 & \xi_1 & \xi_0 \end{bmatrix}. \quad (2.97)$$

因此对于四元数表示 2.51 来说, 映射矩阵 \mathbf{E}_R 的逆是:

$$\mathbf{E}_{R \text{ quat}} = 2\mathbf{H}(\boldsymbol{\xi}) \quad (2.98)$$

$$\mathbf{E}_{R \text{ quat}}^{-1} = \frac{1}{2}\mathbf{H}(\boldsymbol{\xi})^T \quad (2.99)$$

轴角的时间导数 \Leftrightarrow 角速度

对于轴角, 以下关系成立:

$${}^I\boldsymbol{\omega}_{IB} = \mathbf{n}\dot{\theta} + [\mathbf{n}]_{\times}\dot{\mathbf{n}}(1 - \cos\theta) \quad (2.100)$$

$$\dot{\theta} = \mathbf{n}^T {}_I\boldsymbol{\omega}_{IB} \quad \dot{\mathbf{n}} = \left(-\frac{1}{2} \frac{\sin \theta}{1 - \cos \theta} [\mathbf{n}]_{\times}^2 - \frac{1}{2} [\mathbf{n}]_{\times} \right) {}_I\boldsymbol{\omega}_{IB} \quad \forall \theta \in \mathbb{R} \setminus \{0\} \quad (2.101)$$

因此对于轴角 2.46 来说, 映射矩阵 \mathbf{E}_R 和它的逆是:

$$\mathbf{E}_{R \text{ angleaxis}} = \begin{bmatrix} \mathbf{n} & \sin \theta \mathbb{I}_{3 \times 3} + [1 - \cos \theta](\mathbf{n})_{\times} \end{bmatrix} \quad (2.102)$$

$$\mathbf{E}_{R \text{ angleaxis}}^{-1} = \begin{bmatrix} \mathbf{n}^T \\ -\frac{1}{2} \frac{\sin \theta}{1 - \cos \theta} [\mathbf{n}]_{\times}^2 - \frac{1}{2} [\mathbf{n}]_{\times} \end{bmatrix} \quad (2.103)$$

旋转向量的时间导数 \Leftrightarrow 角速度

对于旋转向量, 可以证明以下关系成立:

$${}_I\boldsymbol{\omega}_{IB} = \left[\mathbb{I}_{3 \times 3} + [\boldsymbol{\varphi}]_{\times} \left(\frac{1 - \cos \|\boldsymbol{\varphi}\|}{\|\boldsymbol{\varphi}\|^2} \right) + [\boldsymbol{\varphi}]_{\times}^2 \left(\frac{\|\boldsymbol{\varphi}\| - \sin \|\boldsymbol{\varphi}\|}{\|\boldsymbol{\varphi}\|^3} \right) \right] \dot{\boldsymbol{\varphi}} \quad \forall \|\boldsymbol{\varphi}\| \in \mathbb{R} \setminus \{0\} \quad (2.104)$$

$$\dot{\boldsymbol{\varphi}} = \left[\mathbb{I}_{3 \times 3} - \frac{1}{2} [\boldsymbol{\varphi}]_{\times} + [\boldsymbol{\varphi}]_{\times}^2 \frac{1}{\|\boldsymbol{\varphi}\|^2} \left(1 - \frac{\|\boldsymbol{\varphi}\|}{2} \frac{\sin \|\boldsymbol{\varphi}\|}{1 - \cos \|\boldsymbol{\varphi}\|} \right) \right] {}_I\boldsymbol{\omega}_{IB} \quad \forall \|\boldsymbol{\varphi}\| \in \mathbb{R} \setminus \{0\} \quad (2.105)$$

因此, 旋转向量 2.47 的映射矩阵 \mathbf{E}_R 和其逆是:

$$\mathbf{E}_{R \text{ rotationvector}} = \left[\mathbb{I}_{3 \times 3} + [\boldsymbol{\varphi}]_{\times} \left(\frac{1 - \cos \|\boldsymbol{\varphi}\|}{\|\boldsymbol{\varphi}\|^2} \right) + [\boldsymbol{\varphi}]_{\times}^2 \left(\frac{\|\boldsymbol{\varphi}\| - \sin \|\boldsymbol{\varphi}\|}{\|\boldsymbol{\varphi}\|^3} \right) \right] \quad (2.106)$$

$$\mathbf{E}_{R \text{ rotationvector}}^{-1} = \left[\mathbb{I}_{3 \times 3} - \frac{1}{2} [\boldsymbol{\varphi}]_{\times} + [\boldsymbol{\varphi}]_{\times}^2 \frac{1}{\|\boldsymbol{\varphi}\|^2} \left(1 - \frac{\|\boldsymbol{\varphi}\|}{2} \frac{\sin \|\boldsymbol{\varphi}\|}{1 - \cos \|\boldsymbol{\varphi}\|} \right) \right] \quad (2.107)$$

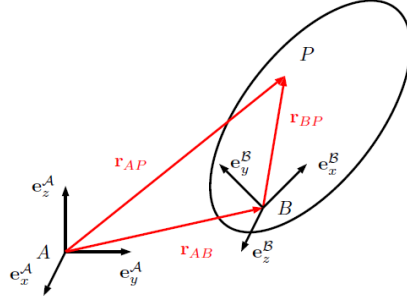
2.6 变换

在最一般的情况下, 两个参考坐标系具有位置偏移和相对旋转 (见图 2.9)。可以使用包含平移和旋转的齐次变换矩阵 \mathbf{T} 将点 P 从一个坐标系转换到另一个坐标系:

$$\mathbf{r}_{AP} = \mathbf{r}_{AB} + \mathbf{r}_{BP} \quad (2.108)$$

$$\mathcal{A}\mathbf{r}_{AP} = \mathcal{A}\mathbf{r}_{AB} + \mathcal{A}\mathbf{r}_{BP} = \mathcal{A}\mathbf{r}_{AB} + \mathbf{C}_{AB} \cdot \mathcal{B}\mathbf{r}_{BP} \quad (2.109)$$

$$\begin{pmatrix} \mathcal{A}\mathbf{r}_{AP} \\ 1 \end{pmatrix} = \underbrace{\begin{bmatrix} \mathbf{C}_{AB} & \mathcal{A}\mathbf{r}_{AB} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}}_{\mathbf{T}_{AB}} \begin{pmatrix} \mathcal{B}\mathbf{r}_{BP} \\ 1 \end{pmatrix} \quad (2.110)$$

图 2.9: 固定在刚体上的坐标系 B

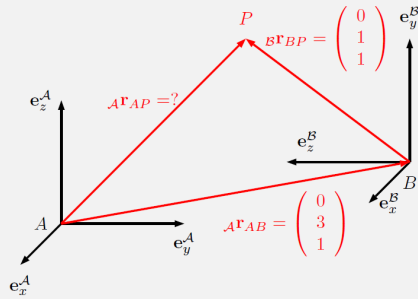
齐次变换的逆可以根据以下公式计算：

$$\mathbf{T}_{AB}^{-1} = \begin{bmatrix} \mathbf{C}_{AB}^T & \overbrace{-\mathbf{C}_{AB}^T \mathbf{r}_{AB}}^{\mathbf{B}^T \mathbf{r}_{BA}} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \quad (2.111)$$

连续的齐次变换由以下公式计算得到：

$$\mathbf{T}_{AC} = \mathbf{T}_{AB} \mathbf{T}_{BC} \quad (2.112)$$

【实例 2.6.1】齐次变换



依据上图中的描述，尝试求出齐次变换矩阵 \mathbf{T}_{AB} 和位置向量 ${}^A\mathbf{r}_{AP}$ 。

齐次变换矩阵是：

$$\mathbf{T}_{AB} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 3 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.113)$$

相应地，位置向量可以通过计算得到：

$$\begin{pmatrix} \mathcal{A}\mathbf{r}_{AP} \\ 1 \end{pmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 3 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \\ 2 \\ 1 \end{pmatrix} \quad (2.114)$$

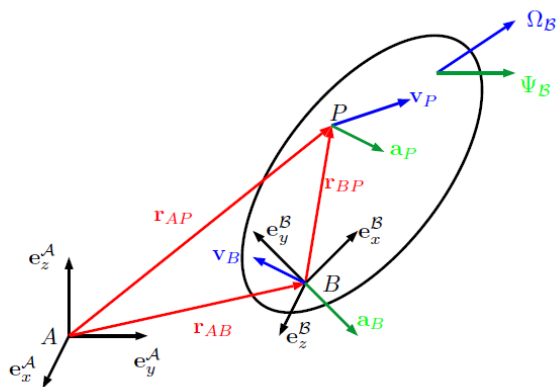


图 2.10: 具有速度和加速度的刚性体

2.7 移动刚体的速度

图 2.10 描绘了单个主体的速度。假设在惯性固定坐标系 \mathcal{A} 中，我们有以下定义：

- \mathbf{v}_P : 点 P 的绝对速度
- $\mathbf{a}_P = \dot{\mathbf{v}}_P$: 点 P 的绝对加速度

- $\Omega_B = \omega_{AB}$: 刚体 B 的 (绝对) 角速度
- $\Psi_B = \dot{\Omega}_B$: 刚体 B 的 (绝对) 角加速度

此时, 重要的是要了解速度之间的区别, 比如说在在坐标系 C 中, 位置相对于绝对时间的变化为:

$${}_C(\dot{\mathbf{r}}_{AP}) = {}_C\left(\frac{d}{dt}\mathbf{r}_{AP}\right) = {}_C\mathbf{v}_{AP} \quad (2.115)$$

以及位置向量坐标相对于时间的微分:

$$({}_C\dot{\mathbf{r}}_{AP}) = ({}_C\mathbf{r}_{AP})' = \frac{d}{dt}({}_C\mathbf{r}_{AP}) \quad (2.116)$$

如果 C 是惯性坐标系, 那么上面两式的相等关系将成立。按照之前介绍的转换, 我们可以将点 P 的位置写为:

$${}_A\mathbf{r}_{AP} = {}_A\mathbf{r}_{AB} + {}_A\mathbf{r}_{BP} = {}_A\mathbf{r}_{AB} + \mathbf{C}_{AB} \cdot {}_B\mathbf{r}_{BP} \quad (2.117)$$

对上面的等式两边对时间求导:

$${}_A\dot{\mathbf{r}}_{AP} = {}_A\dot{\mathbf{r}}_{AB} + \mathbf{C}_{AB} \cdot {}_B\dot{\mathbf{r}}_{BP} + \dot{\mathbf{C}}_{AB} \cdot {}_B\mathbf{r}_{BP} \quad (2.118)$$

因为 P 是刚体上 B 的一个点, 所以相对速度 ${}_B\dot{\mathbf{r}}_{BP} = 0$ 。此外, 从公式 2.72 可以得到 $\dot{\mathbf{C}}_{AB} = [{}_A\omega_{AB}]_{\times} \cdot \mathbf{C}_{AB}$, 于是:

$${}_A\dot{\mathbf{r}}_{AP} = {}_A\dot{\mathbf{r}}_{AB} + [{}_A\omega_{AB}]_{\times} \cdot \mathbf{C}_{AB} \cdot {}_B\mathbf{r}_{BP} \quad (2.119)$$

$$= {}_A\dot{\mathbf{r}}_{AB} + {}_A\omega_{AB} \times {}_A\mathbf{r}_{BP} \quad (2.120)$$

这是非常著名的刚体速度公式, 也被称为速度合成规则, 它可以被重新写成:

$$\mathbf{v}_P = \mathbf{v}_B + \boldsymbol{\Omega} \times \mathbf{r}_{BP} \quad (2.121)$$

对加速度应用相同的计算规则会得到:

$$\begin{aligned} {}_A\ddot{\mathbf{r}}_{AP} &= {}_A\ddot{\mathbf{r}}_{AB} + {}_A\dot{\omega}_{AB} \times {}_A\mathbf{r}_{BP} + {}_A\omega_{AB} \times {}_A\dot{\mathbf{r}}_{BP} \\ &= {}_A\ddot{\mathbf{r}}_{AB} + {}_A\dot{\omega}_{AB} \times {}_A\mathbf{r}_{BP} + {}_A\omega_{AB} \times ({}_A\omega_{AB} \times {}_A\mathbf{r}_{BP}) \\ &\quad \downarrow \\ \mathbf{a}_P &= \mathbf{a}_B + \boldsymbol{\Psi} \times \mathbf{r}_{BP} + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \mathbf{r}_{BP}) \end{aligned} \quad (2.122)$$

关于向量微分的一些注意事项

在运动坐标系中求解向量的微分必须十分小心。需要特别注意的是，速度有时候并不等于位置相对于时间的导数：

$$\mathbf{v}_P \neq \dot{\mathbf{r}}_{AP} \quad (2.123)$$

只有在非移动坐标系 \mathcal{A} 中等号才成立：

$${}_{\mathcal{A}}\mathbf{v}_P = {}_{\mathcal{A}}\dot{\mathbf{r}}_{AP} \quad (2.124)$$

如果我们必须在运动坐标系 \mathcal{B} 中表示速度关系，我们需要应用欧拉微分规则

$${}_{\mathcal{B}}\mathbf{v}_P = \mathbf{C}_{BA} \cdot \frac{d}{dt} (\mathbf{C}_{AB} \cdot {}_{\mathcal{B}}\mathbf{r}_{AP}) \quad (2.125)$$

$$= \mathbf{C}_{BA} \cdot \left(\mathbf{C}_{AB} \cdot {}_{\mathcal{B}}\dot{\mathbf{r}}_{AP} + \dot{\mathbf{C}}_{AB} \mathbf{r}_{AP} \right) \quad (2.126)$$

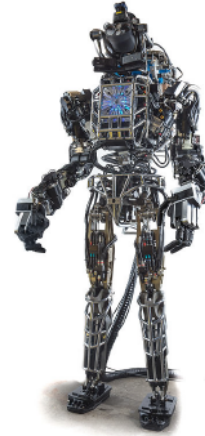
$$= \mathbf{C}_{BA} \cdot \left(\mathbf{C}_{AB} \cdot {}_{\mathcal{B}}\dot{\mathbf{r}}_{AP} + [\mathcal{A}\boldsymbol{\omega}_{AB}]_{\times} \cdot \mathbf{C}_{AB} \cdot {}_{\mathcal{B}}\mathbf{r}_{AP} \right) \quad (2.127)$$

$$= {}_{\mathcal{B}}\dot{\mathbf{r}}_{AP} + \mathbf{C}_{BA} \cdot [\mathcal{A}\boldsymbol{\omega}_{AB}]_{\times} \cdot \mathbf{C}_{AB} \cdot {}_{\mathcal{B}}\mathbf{r}_{AP} \quad (2.128)$$

$$\stackrel{(2.75)}{=} {}_{\mathcal{B}}\dot{\mathbf{r}}_{AP} + {}_{\mathcal{B}}\boldsymbol{\omega}_{AB} \times {}_{\mathcal{B}}\mathbf{r}_{AP} \quad (2.129)$$



(a) ABB IRB 120
6DOF robot arm.



(b) Atlas Humanoid
robot.

图 2.11: 固定基座和浮动基座系统

2.8 刚体系统的运动学

大多数机器人系统可以被建模成由 n_j 个平移或旋转关节连接的 $n_l = n_j + 1$ 个连杆组成的开环运动链，其中每个关节对应于一个自由度。因为在两个连续的刚体之间有单一关节，其位移可以表示成 \mathbf{q}_i ，所以我们可以通过一个简单的变换将两个刚体联系起来：

$$\mathbf{T}_{\mathcal{B}_{i-1}\mathcal{B}_i} = \mathbf{T}_{\mathcal{B}_{i-1}\mathcal{B}_i}(\mathbf{q}_i) \quad (2.130)$$

我们讨论的系统有两种不同的类型，即固定基座和浮动基座系统，其中根连杆要么连接到地面，要么是自由移动的。在下文中，我们如果需要讨论多体运动学的共性时，我们将重点关注固定基座系统，例如机械手。浮动基座系统将在第 2.10 节中具体介绍。在处理固定基座系统时，通常选择附着在根链上的坐标系为固定的世界（惯性）坐标系。

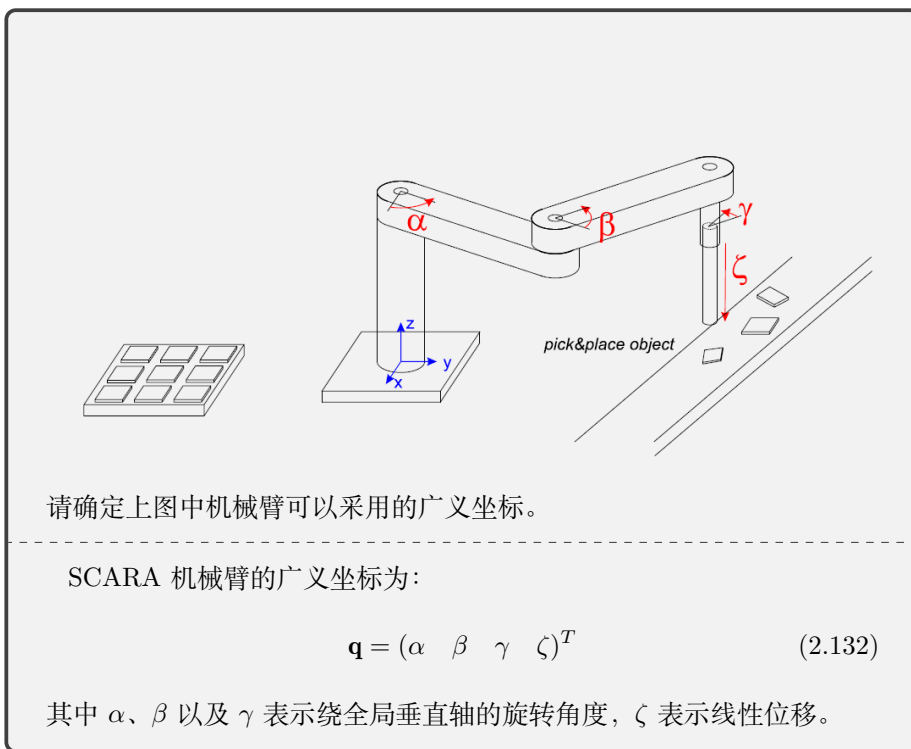
2.8.1 广义坐标和关节位形

机器人（如机械臂）的位形可以用广义坐标向量来描述：

$$\mathbf{q} = \begin{pmatrix} \mathbf{q}_1 \\ \vdots \\ \mathbf{q}_n \end{pmatrix} \quad (2.131)$$

这组标量值必须能够完全描述系统的位形，即对于常数值 \mathbf{q} ，机器人不能发生移动。在大多数情况下，我们将选择互相独立的坐标来作为广义坐标，这也就意味着广义坐标的数量对应于系统自由度。对于没有额外运动约束的固定基座系统，这种最小广义坐标集被称为最小坐标。理解广义坐标的选择不是唯一的是很重要的，不过在大多数应用中，广义坐标对应于机器人的自由度，即对于旋转关节来说，单一自由度 \mathbf{q}_i 对应于关节的旋转角度；如果是移动关节， \mathbf{q}_i 表示线性位移。

【实例 2.8.1】广义坐标和关节位形



2.8.2 任务空间坐标

如图 2.12 所示的机械臂的末端执行器的位形可以通过其相对于参考坐标系的相对位置和方向来描述。参考坐标系通常选惯性或根坐标系。

末端执行器位形参数

正如我们在第 2.2.1 节和第 2.4.5 节中看到的，相对于基坐标系的位置 $\mathbf{r}_e \in \mathbb{R}^3$ 和旋转 $\phi_e \in SO(3)$ 可以通过参数 χ_P 和 χ_R 分别表示。因此，末端执行器的组合位置和方向由以下表示：

$$\mathbf{x}_e = \begin{pmatrix} \mathbf{r}_e \\ \phi_e \end{pmatrix} \in SE(3) \quad (2.133)$$

参数化形式为：

$$\chi_e = \begin{pmatrix} \chi_{eP} \\ \chi_{eR} \end{pmatrix} = \begin{pmatrix} \chi_1 \\ \vdots \\ \chi_m \end{pmatrix} \in \mathbb{R}^m \quad (2.134)$$

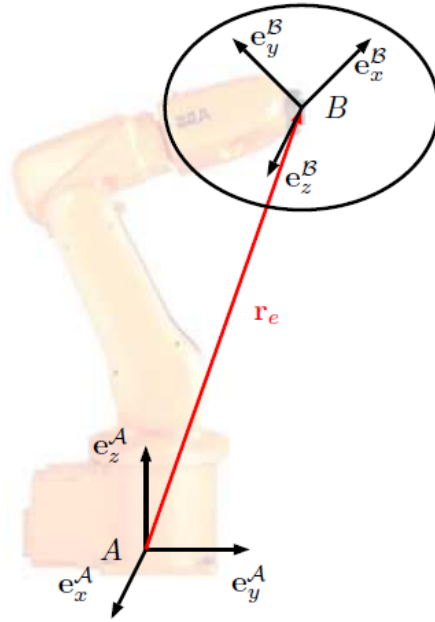


图 2.12: 任务空间坐标为机械臂末端执行器的实例

数字 m 因选择的参数化方法而有所不同。请记住此时旋转 ϕ_e 只是方向的理论抽象，不存在诸如“角位置”之类的数值上的等价物（请参阅第 2.4 节）。

【实例 2.8.2】末端执行器位形

我们可以使用三个笛卡尔位置参数和三个欧拉角参数描述 3D 空间中的末端执行器，则此时需要 $m = 6$ 个参数。如果使用 3 个球坐标位置参数与旋转矩阵的 9 个元素， $m = 12$ 个参数将是必须的。

操作空间坐标

机械臂的末端执行器在所谓的操作空间中操作，操作空间取决于臂的几何形状和结构。操作空间可以使用参数描述为：

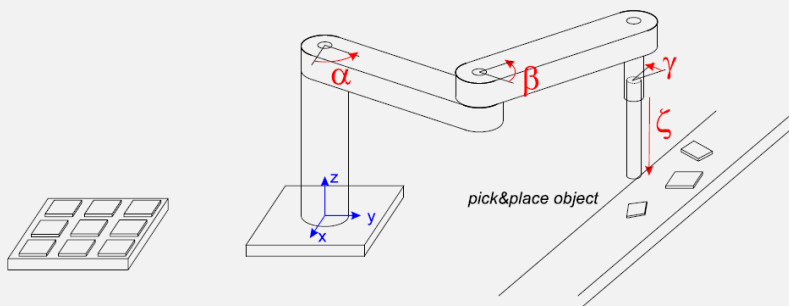
$$\chi_o = \begin{pmatrix} \chi_{oP} \\ \chi_{oR} \end{pmatrix} = \begin{pmatrix} \chi_1 \\ \vdots \\ \chi_{m_0} \end{pmatrix} \quad (2.135)$$

其中 $\chi_1, \chi_2, \dots, \chi_{m_0}$ 是独立的操作空间坐标²，因此，它们可以被理解为末端执行器位形参数的最小选择方式。请注意 $m_0 \leq n_j$ ，这是由于末端执行器处的自由度肯定不大于系统中关节的数量。

【实例 2.8.3】操作空间坐标 1

最一般情况下，在六维操作空间描述末端执行器需要 $m_0 = 6$ 个参数。因此，只有欧拉角是有效的参数化，而不能选择四元数或旋转矩阵。

【实例 2.8.4】操作空间坐标 2



对于上图所示的 SCARA 机械臂来说，操作空间仅为 4 自由度，对应于三个位置自由度和绕垂直轴的旋转自由度。

$$\chi_o = \begin{pmatrix} x \\ y \\ z \\ \phi \end{pmatrix} \quad (2.136)$$

注意：在 SCARA 机械臂的实例中，为 4 自由度机械臂定义四个操作空间坐标非常简单。不过对于具有四个非共线的旋转轴的机械臂来说，选择四个操作空间坐标是不可能的。在这样一个例子中，操作空间坐标仍然是一个相当理论的概念。

²请注意，文献中对操作空间坐标有不同的定义，有些将它们等效于末端坐标。我们在这里将其作为最小表示来方便定义诸如奇点之类的概念。

在下文中，我们将不使用操作空间坐标，而将重点放在末端执行器位形参数的更通用概念上。

2.8.3 正向运动学

正向运动学描述关节广义坐标 \mathbf{q} 和末端执行器位形 χ_e 之间的映射：

$$\chi_e = \chi_e(\mathbf{q}) \quad (2.137)$$

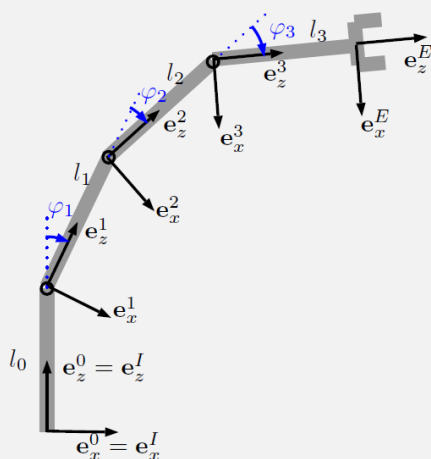
这种关系可以通过分析从基部到末端执行器的运动学公式 2.130 来获得。对于有 n_j 个关节的串连系统来说：

$$\mathbf{T}_{I\mathcal{E}}(\mathbf{q}) = \mathbf{T}_{I0} \cdot \left(\prod_{k=1}^{n_j} \mathbf{T}_{k-1\ k}(q_k) \right) \cdot \mathbf{T}_{n_j\mathcal{E}} = \begin{bmatrix} \mathbf{C}_{I\mathcal{E}}(\mathbf{q}) & {}^I\mathbf{r}_{IE}(\mathbf{q}) \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \quad (2.138)$$

当谈到固定基座机器人时，机器人的第一坐标系 0 是不相对于惯性坐标系移动的，这就使得 \mathbf{T}_{I0} 是一个恒定的变换。此外，在大多数情况下，末端执行器坐标系 \mathcal{E} 与最后一个关节刚性连接，但不必与最后一个坐标系重合。因此 $\mathbf{T}_{n_j\mathcal{E}}$ 也是恒定的。

为了得到类似于公式 2.137 的形式，我们有必要将旋转矩阵 $\mathbf{C}_{I\mathcal{E}}(\mathbf{q})$ 和位置向量 ${}^I\mathbf{r}_{IE}(\mathbf{q})$ 变换成由末端执行器参数 χ_e 表示的。虽然这对于位置来说是直接的，即 $\chi_{eP}(\mathbf{q}) = {}^I\mathbf{r}_{IE}$ ，但是旋转矩阵 $\mathbf{C}_{I\mathcal{E}}(\mathbf{q})$ 的转换根据参数化的选择不同，难度也各不相同（请参阅第 2.5.1 节）。

【实例 2.8.5】正向运动学



找到平面 3 自由度机械臂的正向运动学。

平面 3 自由度机械臂的广义坐标是：

$$\mathbf{q} = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} \varphi_1 \\ \varphi_2 \\ \varphi_3 \end{pmatrix} \quad (2.139)$$

利用这一点，我们计算末端执行器的位置和方向：

$$\chi_e(\mathbf{q}) = \begin{pmatrix} \chi_{eP}(\mathbf{q}) \\ \chi_{eR}(\mathbf{q}) \end{pmatrix} \quad (2.140)$$

$$\begin{aligned} \chi_{eP}(\mathbf{q}) &= \begin{pmatrix} x \\ z \end{pmatrix} \\ &= \begin{pmatrix} l_1 \sin(q_1) + l_2 \sin(q_1 + q_2) + l_3 \sin(q_1 + q_2 + q_3) \\ l_0 + l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) + l_3 \cos(q_1 + q_2 + q_3) \end{pmatrix} \end{aligned} \quad (2.141)$$

$$\chi_{eR}(\mathbf{q}) = \chi_{eR}(\mathbf{q}) = q_1 + q_2 + q_3 \quad (2.142)$$

2.8.4 微分运动学和解析雅可比

通常我们对局部变化或相对于时间的局部变化感兴趣（即速度），这个被称为微分或瞬时运动学。一种常见的方法是线性化正向运动学：

$$\chi_e + \delta\chi_e = \chi_e(\mathbf{q} + \delta\mathbf{q}) = \chi_e(\mathbf{q}) + \frac{\partial\chi_e(\mathbf{q})}{\partial\mathbf{q}}\delta\mathbf{q} + O(\delta\mathbf{q}^2) \quad (2.143)$$

可以得到一阶近似：

$$\delta\chi_e \approx \frac{\partial\chi_e(\mathbf{q})}{\partial\mathbf{q}}\delta\mathbf{q} = \mathbf{J}_{eA}(\mathbf{q})\delta\mathbf{q} \quad (2.144)$$

其中

$$\mathbf{J}_{eA}(\mathbf{q}) = \begin{bmatrix} \frac{\partial\chi_1}{\partial q_1} & \cdots & \frac{\partial\chi_1}{\partial q_{n_j}} \\ \vdots & \ddots & \vdots \\ \frac{\partial\chi_m}{\partial q_1} & \cdots & \frac{\partial\chi_m}{\partial q_{n_j}} \end{bmatrix} \quad (2.145)$$

是 $m \times n_j$ 维的解析雅可比矩阵。雅可比矩阵经常用于机器人系统的运动学和动力学。它将关节的变化与位形空间之间的变化联系到一起，代表了有限差异背景下的一个近似值：

$$\Delta\chi_e \approx \mathbf{J}_{eA}(\mathbf{q})\Delta\mathbf{q} \quad (2.146)$$

从而得到速度之间的精确关系：

$$\dot{\chi}_e = \mathbf{J}_{eA}(\mathbf{q})\dot{\mathbf{q}} \quad (2.147)$$

位置和旋转雅可比

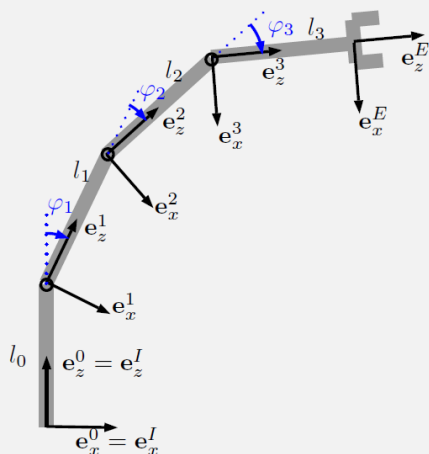
由公式 2.134 指定的末端执行器位形可由末端执行器位置 χ_{eP} 和方向 χ_{eR} 的参数化向量堆叠而成，我们可以将其分为位置和旋转雅可比：

$$\mathbf{J}_{eA} = \begin{bmatrix} \mathbf{J}_{eAP} \\ \mathbf{J}_{eAR} \end{bmatrix} = \begin{bmatrix} \frac{\partial \chi_{eP}}{\partial \mathbf{q}} \\ \frac{\partial \chi_{eR}}{\partial \mathbf{q}} \end{bmatrix} \quad (2.148)$$

对参数化的依赖

正如我们在第 2.2.1 小节和第 2.4.5 小节中看到的，这种雅可比矩阵的维度强烈依赖于所选的参数化方法。例如，当使用欧拉角时， \mathbf{J}_{eAR} 的大小是 $3 \times n_j$ ；如果采用的是四元数，它的大小是 $4 \times n_j$ ；如果采用完整的旋转矩阵作为参数，对应的雅可比矩阵的大小是 $9 \times n_j$ 。

【实例 2.8.6】解析雅可比



找到平面 3 自由度机械臂末端执行器的分析位置和旋转雅可比矩阵。

公式 2.141 求出的 $\chi_{eP}(\mathbf{q})$ 关于 2.139 中的广义坐标 \mathbf{q} 的微分是：

$$\mathbf{J}_{eAP}(\mathbf{q}) = \frac{\partial \chi_{eP}}{\partial \mathbf{q}}$$

$$= \begin{bmatrix} l_1 c_1 + l_2 c_{12} + l_3 c_{123} & l_2 c_{12} + l_3 c_{213} & l_3 c_{213} \\ -l_1 s_1 - l_2 s_{12} - l_3 s_{123} & -l_2 s_{12} - l_3 s_{213} & -l_3 s_{213} \end{bmatrix} \in \mathbb{R}^{2 \times 3} \quad (2.149)$$

其中 $c_{123} = \cos(q_1 + q_2 + q_3)$ 和 $s_{123} = \sin(q_1 + q_2 + q_3)$ 。公式 2.142 给定的 $\chi_{eR}(\mathbf{q})$ 关于 2.139 中广义坐标 \mathbf{q} 的微分方程由下式给出的：

$$\mathbf{J}_{eAR}(\mathbf{q}) = \frac{\partial \chi_{eR}}{\partial \mathbf{q}} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \in \mathbb{R}^{1 \times 3} \quad (2.150)$$

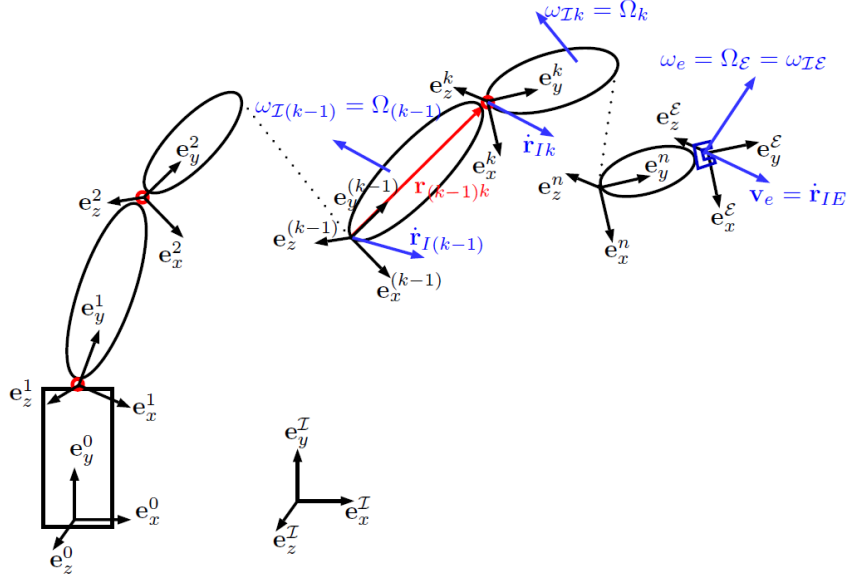


图 2.13: n 个移动连杆的串行连杆臂

2.8.5 几何与基本雅可比

正如我们在公式 2.147 中看到的，雅可比将广义速度（在关节空间中）映射到末端执行器位形表示的时间导数（不是线性和角速度！）。末端执行器位形的相关偏微分 $\mathbf{J}_{eA} = \frac{\partial \chi_e}{\partial \mathbf{q}}$ 取决于所选的参数化方式，尤其是旋转的参数化方式。

然而，正如我们之前所了解到的，刚体具有唯一的线速度 \mathbf{v}_e 和角速度 $\boldsymbol{\omega}_e$ 。因此，必须存在一个唯一的雅可比关系，它将广义速度 $\dot{\mathbf{q}}$ 与末端执行器的速度

(由线速度 \mathbf{v}_e 和角速度 $\boldsymbol{\omega}_e$ 组成) 联系到一起:

$$\mathbf{w}_e = \begin{pmatrix} \mathbf{v}_e \\ \boldsymbol{\omega}_e \end{pmatrix} = \mathbf{J}_{e0}(\mathbf{q})\dot{\mathbf{q}} \quad (2.151)$$

\mathbf{J}_{e0} 被称为几何 (或基本) 雅可比, 在最一般的情况下, 它的维度是 $6 \times n_j$ 。基坐标系 \mathcal{A} 的几何雅可比将广义速度映射到在特定坐标系中表示的末端执行器的速度:

$${}^{\mathcal{A}}\mathbf{w}_e = {}^{\mathcal{A}}\mathbf{J}_{e0}(\mathbf{q})\dot{\mathbf{q}} \quad (2.152)$$

几何雅可比的加法和减法

根据基本运动学知识我们知道一个点的速度 C 可以通过另一个点 B 的速度和 B 和 C 之间的相对速度来计算:

$$\begin{aligned} \mathbf{w}_C &= \begin{pmatrix} \mathbf{v}_C \\ \boldsymbol{\omega}_C \end{pmatrix} = \mathbf{w}_B + \mathbf{w}_{BC} \\ \mathbf{J}_C\dot{\mathbf{q}} &= \mathbf{J}_B\dot{\mathbf{q}} + \mathbf{J}_{BC}\dot{\mathbf{q}} \end{aligned} \quad (2.153)$$

由此我们可以确定几何雅可以简单地相加:

$${}^{\mathcal{A}}\mathbf{J}_C = {}^{\mathcal{A}}\mathbf{J}_B + {}^{\mathcal{A}}\mathbf{J}_{BC} \quad (2.154)$$

当然前提是在相同的参考系中表示。

使用刚体公式计算几何雅可比

对图 2.13 中描述的串联连杆进行刚体上点的速度进行分析 (参考公式 2.121)), 可以得到连杆 k 的速度为:

$$\dot{\mathbf{r}}_{Ik} = \dot{\mathbf{r}}_{I(k-1)} + \boldsymbol{\omega}_{\mathcal{I}(k-1)} \times \mathbf{r}_{(k-1)k} \quad (2.155)$$

请再次记住, 对于数值加法来说, 所有向量都在相同的坐标系下表示是至关重要的。当基础坐标系表示为 0, 末端执行器坐标系为 $n+1$ 时, 末端执行器速度可以写成:

$$\dot{\mathbf{r}}_{IE} = \sum_{k=1}^n \boldsymbol{\omega}_{\mathcal{I}k} \times \mathbf{r}_{k(k+1)} \quad (2.156)$$

使用 \mathbf{n}_k 表示关节的旋转轴 k 可以得到:

$$\boldsymbol{\omega}_{(k-1)k} = \mathbf{n}_k \dot{q}_k \quad (2.157)$$

按照之前的结论：

$$\boldsymbol{\omega}_{\mathcal{I}(k)} = \boldsymbol{\omega}_{\mathcal{I}(k-1)} + \boldsymbol{\omega}_{(k-1)k} \quad (2.158)$$

刚体 k 的角速度可以写成：

$$\boldsymbol{\omega}_{\mathcal{I}k} = \sum_{i=1}^k \mathbf{n}_i \dot{q}_i \quad (2.159)$$

将其代入公式 2.156 并重新排列得到：

$$\dot{\mathbf{r}}_{IE} = \sum_{k=1}^n \left(\sum_{i=1}^k (\mathbf{n}_i \dot{q}_i) \times \mathbf{r}_{k(n+1)} \right) \quad (2.160)$$

$$= \sum_{k=1}^n \mathbf{n}_k \dot{q}_k \times \sum_{i=k}^n \mathbf{r}_{i(i+1)} \quad (2.161)$$

$$= \sum_{k=1}^n \mathbf{n}_k \dot{q}_k \times \mathbf{r}_{k(n+1)} \quad (2.162)$$

将其变成矩阵公式得到几何雅可比：

$$\dot{\mathbf{r}}_{IE} = \underbrace{\begin{bmatrix} \mathbf{n}_1 \times \mathbf{r}_{1(n+1)} & \mathbf{n}_2 \times \mathbf{r}_{2(n+1)} & \dots & \mathbf{n}_n \times \mathbf{r}_{n(n+1)} \end{bmatrix}}_{\mathbf{J}_{e0P}} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{pmatrix} \quad (2.163)$$

给定公式 2.159，旋转雅可比是可以表示成：

$$\boldsymbol{\omega}_{\mathcal{IE}} = \sum_{i=1}^n \mathbf{n}_i \dot{q}_i = \underbrace{\begin{bmatrix} \mathbf{n}_1 & \mathbf{n}_2 & \dots & \mathbf{n}_n \end{bmatrix}}_{\mathbf{J}_{e0R}} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{pmatrix} \quad (2.164)$$

结合这两个表达式可以得到组合的几何雅可比：

$$\mathbf{J}_{e0} = \begin{bmatrix} \mathbf{J}_{e0P} \\ \mathbf{J}_{e0R} \end{bmatrix} = \begin{bmatrix} \mathbf{n}_1 \times \mathbf{r}_{1(n+1)} & \mathbf{n}_2 \times \mathbf{r}_{2(n+1)} & \dots & \mathbf{n}_n \times \mathbf{r}_{n(n+1)} \\ \mathbf{n}_1 & \mathbf{n}_2 & \dots & \mathbf{n}_n \end{bmatrix} \quad (2.165)$$

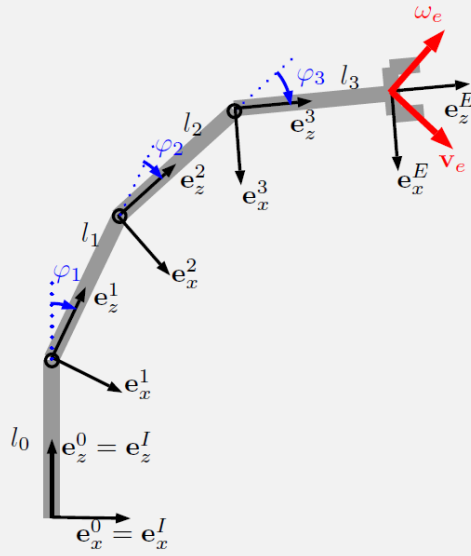
如本节开头所述，重要的是我们需要根据一个基坐标系来定义这个雅可比，例如坐标系 \mathcal{I} (或任何其他坐标系)：

$${}_{\mathcal{I}}\mathbf{J}_{e0} = \begin{bmatrix} {}_{\mathcal{I}}\mathbf{J}_{e0P} \\ {}_{\mathcal{I}}\mathbf{J}_{e0R} \end{bmatrix} = \begin{bmatrix} {}_{\mathcal{I}}\mathbf{n}_1 \times {}_{\mathcal{I}}\mathbf{r}_{1(n+1)} & {}_{\mathcal{I}}\mathbf{n}_2 \times {}_{\mathcal{I}}\mathbf{r}_{2(n+1)} & \dots & {}_{\mathcal{I}}\mathbf{n}_n \times {}_{\mathcal{I}}\mathbf{r}_{n(n+1)} \\ {}_{\mathcal{I}}\mathbf{n}_1 & {}_{\mathcal{I}}\mathbf{n}_2 & \dots & {}_{\mathcal{I}}\mathbf{n}_n \end{bmatrix} \quad (2.166)$$

在此公式中，旋转轴由以下公式给定：

$$\mathcal{I}\mathbf{n}_k = \mathbf{C}_{\mathcal{I}(k-1)} \cdot {}^{(k-1)}\mathbf{n}_k \quad (2.167)$$

【实例 2.8.7】基本雅可比



确定该平面机械臂的基本雅可比。

广义坐标是：

$$\mathbf{q} = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} \varphi_1 \\ \varphi_2 \\ \varphi_3 \end{pmatrix} \quad (2.168)$$

步骤 1: 确定旋转矩阵

$$\mathbf{C}_{\mathcal{I}1} = \begin{bmatrix} c_1 & 0 & s_1 \\ 0 & 1 & 0 \\ -s_1 & 0 & c_1 \end{bmatrix} \quad (2.169)$$

$$\mathbf{C}_{\mathcal{I}2} = \mathbf{C}_{\mathcal{I}1} \cdot \begin{bmatrix} c_2 & 0 & s_2 \\ 0 & 1 & 0 \\ -s_2 & 0 & c_2 \end{bmatrix} = \begin{bmatrix} c_{12} & 0 & s_{12} \\ 0 & 1 & 0 \\ -s_{12} & 0 & c_{12} \end{bmatrix} \quad (2.170)$$

$$\mathbf{C}_{I3} = \mathbf{C}_{I2} \cdot \begin{bmatrix} c_3 & 0 & s_3 \\ 0 & 1 & 0 \\ -s_3 & 0 & c_3 \end{bmatrix} = \begin{bmatrix} c_{123} & 0 & s_{123} \\ 0 & 1 & 0 \\ -s_{123} & 0 & c_{123} \end{bmatrix} \quad (2.171)$$

带 $s_{123} = \sin(q_1 + q_2 + q_3)$ 和 $c_{123} = \cos(q_1 + q_2 + q_3)$.

步骤 2: 确定局部旋转轴 ${}_{(k-1)}\mathbf{n}_k$:

$${}_0\mathbf{n}_1 = {}_1\mathbf{n}_2 = {}_2\mathbf{n}_3 = \mathbf{e}_y \quad (2.172)$$

步骤 3: 确定旋转轴 ${}_I\mathbf{n}_k = \mathbf{C}_{I(k-1)} \cdot {}_{(k-1)}\mathbf{n}_k$:

$${}_I\mathbf{n}_1 = {}_0\mathbf{n}_1 = \mathbf{e}_y \quad (2.173)$$

$${}_I\mathbf{n}_2 = \mathbf{C}_{I1} \cdot {}_1\mathbf{n}_2 = \mathbf{e}_y \quad (2.174)$$

$${}_I\mathbf{n}_3 = \mathbf{C}_{I2} \cdot {}_2\mathbf{n}_3 = \mathbf{e}_y \quad (2.175)$$

步骤 4: 确定从关节到末端执行器的位置向量:

$${}_I\mathbf{r}_{1E} = {}_I\mathbf{r}_{12} + {}_I\mathbf{r}_{23} + {}_I\mathbf{r}_{3E} \quad (2.176)$$

$$= \mathbf{C}_{I1} \cdot {}_1\mathbf{r}_{12} + \mathbf{C}_{I2} \cdot {}_2\mathbf{r}_{23} + \mathbf{C}_{I3} \cdot {}_3\mathbf{r}_{3E} \quad (2.177)$$

$$= l_1 \begin{pmatrix} s_{q_1} \\ 0 \\ c_{q_1} \end{pmatrix} + l_2 \begin{pmatrix} s_{12} \\ 0 \\ c_{12} \end{pmatrix} + l_3 \begin{pmatrix} s_{123} \\ 0 \\ c_{123} \end{pmatrix} \quad (2.178)$$

$${}_I\mathbf{r}_{2E} = {}_I\mathbf{r}_{23} + {}_I\mathbf{r}_{3E} \quad (2.179)$$

$$= \mathbf{C}_{I2} \cdot {}_2\mathbf{r}_{23} + \mathbf{C}_{I3} \cdot {}_3\mathbf{r}_{3E} \quad (2.180)$$

$$= l_2 \begin{pmatrix} s_{12} \\ 0 \\ c_{12} \end{pmatrix} + l_3 \begin{pmatrix} s_{123} \\ 0 \\ c_{123} \end{pmatrix} \quad (2.181)$$

$${}_I\mathbf{r}_{3E} = {}_I\mathbf{r}_{3E} \quad (2.182)$$

$$= \mathbf{C}_{I3} \cdot {}_3\mathbf{r}_{3E} \quad (2.183)$$

$$= l_3 \begin{pmatrix} s_{123} \\ 0 \\ c_{123} \end{pmatrix} \quad (2.184)$$

步骤 5a: 确定雅可比的位置:

$${}_I\mathbf{J}_{e0P} = \begin{bmatrix} {}_I\mathbf{n}_1 \times {}_I\mathbf{r}_{1E} & {}_I\mathbf{n}_2 \times {}_I\mathbf{r}_{2E} & {}_I\mathbf{n}_3 \times {}_I\mathbf{r}_{3E} \end{bmatrix} \quad (2.185)$$

$$= \begin{bmatrix} l_1 c_1 + l_2 c_{12} + l_3 c_{123} & l_2 c_{12} + l_3 c_{123} & l_3 c_{123} \\ 0 & 0 & 0 \\ -l_1 s_1 - l_2 s_{12} - l_3 s_{123} & -l_2 s_{12} - l_3 s_{123} & -l_3 s_{123} \end{bmatrix} \quad (2.186)$$

步骤 5b: 确定旋转雅可比:

$$\mathcal{I}\mathbf{J}_{e0R} = \begin{bmatrix} \mathcal{I}\mathbf{n}_1 & \mathcal{I}\mathbf{n}_2 & \mathcal{I}\mathbf{n}_3 \end{bmatrix} \quad (2.187)$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad (2.188)$$

2.8.6 几何矩阵与解析雅可比矩阵的关系

如第 2.2.1 小节和第 2.5.1 小节所述, 末端执行器表示参数的微分 $\dot{\chi}_e$ 和由线性和角速度组成的旋量 \mathbf{w}_e 之间存在映射。这种关系也体现在位置和旋转相对于广义坐标的偏微分 $\mathbf{J}_{eA} = \frac{\partial \chi_e}{\partial \mathbf{q}} \in \mathbb{R}^{m \times n_j}$ 和几何雅可比 $\mathbf{J}_{e0} \in \mathbb{R}^{6 \times n_j}$ 之间。给定

$$\dot{\chi}_e = \mathbf{J}_{eA}(\mathbf{q})\dot{\mathbf{q}} \quad \text{其中 } \mathbf{J}_{eA}(\mathbf{q}) \in \mathbb{R}^{m_e \times n_j} \quad (2.189)$$

$$\mathbf{w}_e = \mathbf{J}_{e0}(\mathbf{q})\dot{\mathbf{q}} \quad \text{其中 } \mathbf{J}_{e0}(\mathbf{q}) \in \mathbb{R}^{6 \times n_j} \quad (2.190)$$

$$\mathbf{w}_e = \mathbf{E}_e(\chi_e) \dot{\chi}_e \quad \text{其中 } \mathbf{E}_e(\chi_e) = \begin{bmatrix} \mathbf{E}_P & 0 \\ 0 & \mathbf{E}_R \end{bmatrix} \in \mathbb{R}^{m_e \times 6} \quad (2.191)$$

以下映射同样成立:

$$\mathbf{J}_{e0}(\mathbf{q}) = \mathbf{E}_e(\chi)\mathbf{J}_{eA}(\mathbf{q}) \quad (2.192)$$

请注意依赖参数化方式的矩阵 E 和 E^{-1} 是在之前的章节已经求出的:

- 笛卡尔坐标公式 2.12
- 圆柱形坐标公式 2.14 和公式 2.15
- 球形坐标公式 2.16 和公式 2.17
- 欧拉角
 - XYZ : 公式 2.88 和公式 2.89

- ZYX : 公式 2.86 和公式 2.87
- ZYZ : 公式 2.90 和公式 2.91
- 四元数公式 2.98 和公式 2.99
- 轴角公式 2.102 和公式 2.103
- 旋转向量公式 2.106 和公式 2.107

文献中通常不区分几何和解析雅可比。大多数情况下, \mathbf{J} 在文献中指的是几何雅可比, 这门课程也是如此。对于平面系统, 分析和几何雅可比是相同的。

2.9 运动学控制方法

先前引入的末端执行器和关节空间位形之间的关系通常用于运动学控制。

2.9.1 逆微分运动学

正如我们所看到的, 雅可比 $\mathbf{J}_{e0}(q)$ 是关节空间速度 $\dot{\mathbf{q}}$ 和末端执行器速度 \mathbf{w}_e 之间的简单映射:

$$\mathbf{w}_e = \mathbf{J}_{e0} \dot{\mathbf{q}} \quad (2.193)$$

然而, 在许多情况下, 特别是在控制领域, 我们对解决逆问题更加感兴趣: 即给定期望的末端执行器速度 \mathbf{w}_e^* , 求解对应的关节速度 $\dot{\mathbf{q}}$ 。一个显而易见的方法是简单地反转雅可比矩阵, 或者更一般地, 取雅可比的伪逆:

$$\dot{\mathbf{q}} = \mathbf{J}_{e0}^+ \mathbf{w}_e^* \quad (2.194)$$

然而, 正如我们所知, 伪逆的求解存在不同种类的方法, 并且 \mathbf{J}_{e0} 的特性会影响解的唯一性。

奇异性

如果机器人所处的位形 \mathbf{q}_s 对应的雅可比矩阵 $\mathbf{J}_{e0}(\mathbf{q}_s)$ 满足 $\text{rank} < m_0$, 其中 m_0 为操作空间坐标的数量 (即末端执行器的可控自由度), 该位形称为奇位形。奇异位形意味着给定期望的末端执行器速度 \mathbf{w}_e^* 不存在广义速度 $\dot{\mathbf{q}}$ 能够满足等式 $\mathbf{w}_e^* = \mathbf{J}_{e0} \dot{\mathbf{q}}$ 。但可以通过求解摩尔-彭罗斯伪逆, 对应的解 $\dot{\mathbf{q}} = \mathbf{J}_{e0}^+ \mathbf{w}_e^*$ 能够最小化平方误差 $\|\mathbf{w}_e^* - \mathbf{J}_{e0} \dot{\mathbf{q}}\|^2$ 。

不幸的是, 如果机器人接近奇异位形, 即奇异值接近 0 时, \mathbf{J}_{e0} 的条件数变得糟糕。这意味着在相应的方向产生很小的速度 \mathbf{w}_e^* 会需要极高的关节速度 $\dot{\mathbf{q}}$ 。

这种行为对于我们在第 2.9.3 小节中介绍的逆运动学算法的求解带来了很大问题。

一些奇异位形是非常明显的，例如处于其工作空间的边界的机械臂，因此很容易避开。然而，也可能存在虽然在工作区内但某些关节的轴是平行的情况，诸如此类的情況就很难预防，需要对机器人的运动进行谨慎的规划。当使用 6 自由度臂并控制相同数量的末端执行器坐标时，情况尤其如此。一种通常使用的方法是使用具有 7 自由度的机械臂，以防止机器人在跟随所需的期望位姿时卡在奇异位形中。因此，系统的冗余可以在跟踪相同的末端执行器姿态轨迹的同时，使机器人位形远离奇异位形。

阻尼解

为了处理接近奇异的条件数不好的 \mathbf{J}_{e0} ，一个常见的方法是使用带阻尼的摩尔-彭罗斯伪逆：

$$\dot{\mathbf{q}} = \mathbf{J}_{e0}^T (\mathbf{J}_{e0} \mathbf{J}_{e0}^T + \lambda^2 \mathbb{I})^{-1} \mathbf{w}_e^* \quad (2.195)$$

其中 λ 为阻尼参数。阻尼参数越大，所得到的解越稳定，但收敛速度越慢。这种求逆方法使误差 $\|\mathbf{w}_e^* - \mathbf{J}_{e0} \dot{\mathbf{q}}\|^2 + \lambda^2 \|\dot{\mathbf{q}}\|^2$ 最小化。更多详细有关信息，请参阅手稿 [3]。

冗余

如果机器人所处的位形 \mathbf{q}^* 对应的雅可比矩阵 $\mathbf{J}_{e0}(\mathbf{q}^*)$ 满足 $\text{rank} < n$ ，该位形称为冗余。当采用摩尔-彭罗斯伪逆时，如在前面的情况下，我们可以得到解：

$$\dot{\mathbf{q}} = \mathbf{J}_{e0}^T (\mathbf{J}_{e0} \mathbf{J}_{e0}^T)^{-1} \mathbf{w}_e^* \quad (2.196)$$

此时会在满足 $\mathbf{w}_e^* = \mathbf{J}_{e0} \dot{\mathbf{q}}$ 的前提下最小化误差 $\|\dot{\mathbf{q}}\|_2$ 。系统中的冗余意味着存在无限多的解：

$$\dot{\mathbf{q}} = \mathbf{J}_{e0}^+ \mathbf{w}_e^* + \mathbf{N} \dot{\mathbf{q}}_0 \quad (2.197)$$

其中 $\mathbf{N} = \mathcal{N}(\mathbf{J}_{e0})$ 是满足 $\mathbf{J}_{e0} \mathbf{N} = \mathbf{0}$ 的 \mathbf{J}_{e0} 的零空间投影矩阵。这允许我们在不改变末端执行器运动的前提下，通过任意选择 $\dot{\mathbf{q}}_0$ 来修改广义速度 $\dot{\mathbf{q}}$ ：

$$\mathbf{J}_{e0} (\mathbf{J}_{e0}^+ \mathbf{w}_e^* + \mathbf{N} \dot{\mathbf{q}}_0) = \mathbf{w}_e^* \quad \forall \dot{\mathbf{q}}_0 \quad (2.198)$$

存在不同的方法来确定空空间投影矩阵，最简单的投影矩阵如下：

$$\mathbf{N} = \mathbb{I} - \mathbf{J}_{e0}^+ \mathbf{J}_{e0} \quad (2.199)$$

2.9.2 多任务逆微分运动学控制

对于例如跟随特定轨迹、达到末端执行器方向、确保运动学约束等任务或操作空间目标来说，任务雅可比和期望的任务速度可以定义为：

$$\text{task}_i := \{\mathbf{J}_i \mathbf{w}_i^*\} \quad (2.200)$$

相同优先级的多任务

如果所有的 n_t 个任务具有相同的优先级，广义速度可以由以下公式得到：

$$\dot{\mathbf{q}} = \underbrace{\begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_{n_t} \end{bmatrix}}_{\bar{\mathbf{J}}} \underbrace{\begin{pmatrix} \mathbf{w}_1^* \\ \vdots \\ \mathbf{w}_{n_t}^* \end{pmatrix}}_{\bar{\mathbf{w}}} \quad (2.201)$$

如果堆叠雅可比矩阵 $\bar{\mathbf{J}}$ 的行秩大于列秩，此时我们仅需要在最小二乘最优意义上 $\|\bar{\mathbf{w}} - \bar{\mathbf{J}}\dot{\mathbf{q}}\|_2$ 完成任务。使用加权伪逆矩阵能够将特定的任务的优先级设定为高优先级。

$$\bar{\mathbf{J}}^{+W} = \left(\bar{\mathbf{J}}^T \mathbf{W} \bar{\mathbf{J}} \right)^{-1} \bar{\mathbf{J}}^T \mathbf{W} \quad (2.202)$$

其中加权矩阵为 $\mathbf{W} = \text{diag}(w_1 \dots w_m)$ ，对应于误差 $\|\mathbf{W}^{1/2}(\bar{\mathbf{w}} - \bar{\mathbf{J}}\dot{\mathbf{q}})\|_2$ 的最小化。因为 \mathbf{W} 是对角矩阵，Cholesky 因子 $1/2$ 对应于元素的平方根。

具有优先级的多任务

为了明确地保证多任务的优先级，一种方法是使用连续的零空间投影来确定解。假设 n_t 个任务的优先级依次降低，正如我们在公式 2.197 中所看到的， task_1 对应的广义速度是：

$$\dot{\mathbf{q}} = \mathbf{J}_1^+ \mathbf{w}_1^* + \mathbf{N}_1 \dot{\mathbf{q}}_0 \quad (2.203)$$

因此，为了不违背第一个目标，它必须满足：

$$\mathbf{w}_2 = \mathbf{J}_2 \dot{\mathbf{q}} = \mathbf{J}_2 (\mathbf{J}_1^+ \mathbf{w}_1^* + \mathbf{N}_1 \dot{\mathbf{q}}_0) \quad (2.204)$$

这可以求解 $\dot{\mathbf{q}}_0$ ：

$$\dot{\mathbf{q}}_0 = (\mathbf{J}_2 \mathbf{N}_1)^+ (\mathbf{w}_2^* - \mathbf{J}_2 \mathbf{J}_1^+ \mathbf{w}_1^*) \quad (2.205)$$

将其代入公式 2.203 中，得到：

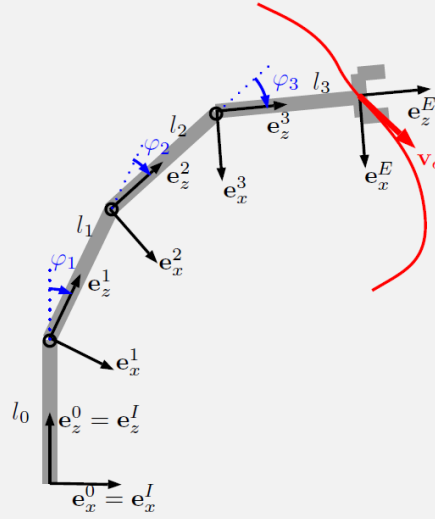
$$\dot{\mathbf{q}} = \mathbf{J}_1^+ \mathbf{w}_1^* + \mathbf{N}_1 (\mathbf{J}_2 \mathbf{N}_1)^+ (\mathbf{w}_2^* - \mathbf{J}_2 \mathbf{J}_1^+ \mathbf{w}_1^*) \quad (2.206)$$

对于 n_t 个任务来说, 解可以以递归方式写做:

$$\dot{\mathbf{q}} = \sum_{i=1}^{n_t} \bar{\mathbf{N}}_i \dot{\mathbf{q}}_i \quad \text{with} \quad \dot{\mathbf{q}}_i = (\mathbf{J}_i \bar{\mathbf{N}}_i)^+ \left(\mathbf{w}_i^* - \mathbf{J}_i \sum_{k=1}^{i-1} \bar{\mathbf{N}}_k \dot{\mathbf{q}}_k \right) \quad (2.207)$$

其中 $\bar{\mathbf{N}}_i$ 是堆叠雅可比矩阵 $\bar{\mathbf{J}}_i = \begin{bmatrix} \mathbf{J}_1^T & \dots & \mathbf{J}_{i-1}^T \end{bmatrix}^T$ 的零空间投影。

【实例 2.9.1】多任务控制



这个例子再次讨论了具有单位连杆长度的 3 自由度平面机械臂。给定定义于 \mathbb{R}^2 上的期望的末端执行器速度 ${}^0\dot{\mathbf{r}}_E^*(t)$ 。时刻 t 时, 机器人处于位形 $\mathbf{q}_t = (\pi/6 \ \pi/3 \ \pi/3)^T$, 期望的末端执行器速度为 ${}^0\dot{\mathbf{r}}_{E\ t}^* = (1 \ 1)^T$ 。

请问末端执行器跟踪问题的解析解是?

期望的末端执行器运动是:

$$\mathbf{w}_E^* = {}^r\dot{\mathbf{r}}_E^* \quad (2.208)$$

使用在实例 2.8.4 和 2.8.5 中计算的雅可比矩阵中的 \mathbf{x} 和 \mathbf{z} 行, 可得到 \mathbf{J}_E , 那么广义速度可以通过以下公式计算得到:

$$\dot{\mathbf{q}} = \mathbf{J}_E^+ \mathbf{w}_E^* \quad (2.209)$$

请问末端执行器跟踪问题的数值解是什么?

这一瞬间的目标速度是:

$$\mathbf{w}_{E t}^* = {}_0\dot{\mathbf{r}}_{E t}^* = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad (2.210)$$

位形 $\mathbf{q}_t = (\pi/6 \ \pi/3 \ \pi/3)$ 对应的雅可比矩阵是:

$$\mathbf{J}_{E t} = \mathbf{J}_E(\mathbf{q}_t) = \frac{1}{2} \begin{bmatrix} 0 & -\sqrt{3} & -\sqrt{3} \\ -4 & -3 & -1 \end{bmatrix} \quad (2.211)$$

按照摩尔-彭罗斯伪逆得到的结果:

$$\dot{\mathbf{q}}_t^{\text{single}} = \mathbf{J}_{E t}^+ \mathbf{w}_{E t}^* = \begin{pmatrix} 0.069 \\ -0.560 \\ -0.595 \end{pmatrix} \quad (2.212)$$

如我们所见, 上述广义速度产生精确的任务空间速度为:

$$\mathbf{w}_{E t} = \mathbf{J}_{E t} \dot{\mathbf{q}}_t^{\text{single}} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \mathbf{w}_{E t}^* \quad (2.213)$$

除了以期望速度移动末端执行器外, 我们还希望第一和第二关节速度为零。如果所有的任务都应该以同等的优先级完成, 那么一般的解是什么?

除了先前定义的任务, 我们还引入了一个附加任务来保持所有关节速度相等:

$$\mathbf{w}_j = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \mathbf{J}_j = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (2.214)$$

堆叠问题现在可以通过以下公式求解:

$$\dot{\mathbf{q}} = \begin{bmatrix} \mathbf{J}_E \\ \mathbf{J}_j \end{bmatrix}^+ \begin{pmatrix} \mathbf{w}_E^* \\ \mathbf{w}_j^* \end{pmatrix} \quad (2.215)$$

堆叠问题的数值解是什么？

$$\dot{\mathbf{q}}_t^{\text{stack}} = \begin{bmatrix} \mathbf{J}_{E\ t} \\ \mathbf{J}_{j\ t} \end{bmatrix}^+ \begin{pmatrix} \mathbf{w}_E^* \\ \mathbf{w}_j^* \end{pmatrix} \quad (2.216)$$

$$= \begin{pmatrix} -0.133 \\ -0.067 \\ -1.132 \end{pmatrix} \quad (2.217)$$

我们注意到这种关节速度会导致：

$$\begin{bmatrix} \mathbf{J}_{E\ t} \\ \mathbf{J}_{j\ t} \end{bmatrix} \dot{\mathbf{q}}_t^{\text{stack}} = \begin{pmatrix} 1.039 \\ 0.933 \\ -0.133 \\ -0.067 \end{pmatrix} \neq \begin{pmatrix} \mathbf{w}_E^* \\ \mathbf{w}_{\Delta\dot{q}}^* \end{pmatrix} \quad (2.218)$$

接下来让我们关注下此时这两个任务的误差，这两个任务的误差分别是

$$\|\mathbf{w}_{E\ t}^* - \mathbf{J}_{E\ t} \dot{\mathbf{q}}_t^{\text{stack}}\|^2 = 0.0059 \quad (2.219)$$

$$\|\mathbf{w}_{j\ t}^* - \mathbf{J}_{j\ t} \dot{\mathbf{q}}_t^{\text{stack}}\|^2 = 0.0223 \quad (2.220)$$

正如我们所看到的，这两项任务都没有被完成。相比之下，没有关节速度约束的解 2.212 的误差为：

$$\|\mathbf{w}_{E\ t}^* - \mathbf{J}_{E\ t} \dot{\mathbf{q}}_t^{\text{single}}\|^2 = 0 \quad (2.221)$$

$$\|\mathbf{w}_{j\ t}^* - \mathbf{J}_{j\ t} \dot{\mathbf{q}}_t^{\text{single}}\|^2 = 0.319 \quad (2.222)$$

公式 2.227 对应的第二个任务中的误差大于关节约束优化的两个任务 2.219 和 2.220 的误差之和。换句话说，当针对两个任务同时进行优化时，为了降低关节 1 和关节 2 中的速度，我们牺牲了末端执行器跟踪性能。

如果将末端执行器跟踪任务设置为前两个关节速度保持为零具有更高的优先级，那么一般解决方案是什么？

为此, 我们使用 2.206 中给出的形式:

$$\dot{\mathbf{q}}_t^{\text{prio}} = \mathbf{J}_1^+ \mathbf{w}_1^* + \mathbf{N}_1 (\mathbf{J}_2 \mathbf{N}_1)^+ (\mathbf{w}_2^* - \mathbf{J}_2 \mathbf{J}_1^+ \mathbf{w}_1^*) \quad (2.223)$$

其中 $\mathbf{J}_1 = \mathbf{J}_{E t}$, $\mathbf{J}_2 = \mathbf{J}_{j t}$, $\mathbf{w}_1^* = \mathbf{w}_{E t}^*$ 和 $\mathbf{w}_2 = \mathbf{w}_{j t}^*$ 。一个可能的零空间投影矩阵是:

$$\mathbf{N}_1 = \mathbb{I} - \mathbf{J}_1^+ \mathbf{J}_1 = \frac{1}{9} \begin{bmatrix} 1 & -2 & 2 \\ -2 & 4 & -4 \\ 2 & -4 & 4 \end{bmatrix} \quad (2.224)$$

使用 \mathbf{N}_1 , 其 $\text{rank} = 1$, 对应的广义速度为:

$$\dot{\mathbf{q}}_t^{\text{prio}} = \begin{pmatrix} -0.169 - 0.085 & -1.070 \end{pmatrix} \quad (2.225)$$

同样, 让我们看看误差:

$$\|\mathbf{w}_{E t}^* - \mathbf{J}_{E t} \dot{\mathbf{q}}_t^{\text{prio}}\|^2 = 0 \quad (2.226)$$

$$\|\mathbf{w}_{j t}^* - \mathbf{J}_{j t} \dot{\mathbf{q}}_t^{\text{prio}}\|^2 = 0.036 \quad (2.227)$$

虽然与堆叠解决方案相比, 总误差较大, 但末端执行器跟踪任务被准确执行了。与单个任务优化相比, 可以看出总误差减小了。实际上, 使用一维零空间, 能够在最佳意义上在不改变末端执行器跟踪性能的情况下降低关节速度。

2.9.3 逆运动学

逆运动学的目标是找到一个关节位形 \mathbf{q} , 其是给定末端执行器位形 χ_e^* 的函数:

$$\mathbf{q} = \mathbf{q}(\chi_e^*) \quad (2.228)$$

解析解

对于一大类机械结构来说, 是存在解析解的, 事实上, 许多机械臂被特别设计成可以解析地求解逆运动学问题。对于经典的 6 自由度机械臂来说, 解析解存在的必要条件是三个相邻轴相交。解析求解逆运动学问题有两种被广泛使用的方法: 几何方法是将机械手的空间几何分解成几个平面问题, 并应用几何定律进行求解; 代数方法是操纵变换矩阵来得到角度。有关这个话题的更多信息,

请感兴趣的读者参考 [8]。

数值解

随着计算能力的提高，数值方法成为解决逆运动学问题的常用工具。正如我们在第 2.147 小节中看到的，可以使用解析雅可比直接将关节空间坐标的差 $\Delta \mathbf{q}$ 映射成末端执行器坐标的差 $\Delta \mathbf{x}_e$ ：

$$\Delta \mathbf{x}_e = \mathbf{J}_{eA} \Delta \mathbf{q} \quad (2.229)$$

这种关系可以在给定初始位形 \mathbf{q}^0 和期望的末端执行器位形 \mathbf{x}_e^* 的条件下，迭代地求解逆运动学问题（参见算法 2.9.3）。该算法迭代完成以下步骤，直到达到满足 $\|\Delta \mathbf{x}_e\| = \|\mathbf{x}_e^* - \mathbf{x}_e(\mathbf{q}^*)\| < tol$ ，此时到达目标位置 \mathbf{q}^* 。

算法 1 数值逆运动学

```

1 :  $\mathbf{q} \leftarrow \mathbf{q}^0$                                 ▷ 初始位形
2 : while  $\|\mathbf{x}_e^* - \mathbf{x}_e(\mathbf{q})\| > tol$  do                ▷ 未求到解
3 :    $\mathbf{J}_{eA} \leftarrow \mathbf{J}_{eA}(\mathbf{q}) = \frac{\partial \mathbf{x}_e}{\partial \mathbf{q}}(\mathbf{q})$     ▷ 评估位形  $\mathbf{q}$  的雅可比
4 :    $\mathbf{J}_{eA}^+ \leftarrow (\mathbf{J}_{eA})^+$                     ▷ 计算伪逆
5 :    $\Delta \mathbf{x}_e \leftarrow \mathbf{x}_e^* - \mathbf{x}_e(\mathbf{q})$               ▷ 计算末端执行器位形误差向量
6 :    $\mathbf{q} \leftarrow \mathbf{q} + \mathbf{J}_{eA}^+ \Delta \mathbf{x}_e$             ▷ 更新广义坐标
7 : end while

```

不幸的是，这种方法同样存在一些问题。首先，如果解析雅可比实现的误差线性化不够准确，目标和实际位形之间存在的误差 $\Delta \mathbf{x}_e^i$ 将越来越大。解决此问题的一种简单方法是放大每个更新步骤：

$$\mathbf{q} \leftarrow \mathbf{q} + k \mathbf{J}_{eA}^+ \Delta \mathbf{x}_e \quad (2.230)$$

其中 $0 < k < 1$ ，是为了保持依然在线性化的有效范围内，避免超调或发散。然而，这导致收敛速度较慢。当处理非常小的 k 增益时，该解跟随由解析雅可比表示的局部线性化，直到完全收敛。

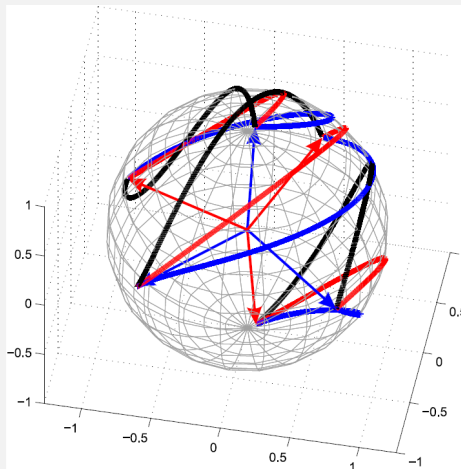
其次，如果目标接近奇异位置，雅可比的逆将成为一个差条件数的问题。如第 2.9.1 小节所示，此时可以使用带阻尼的逆来克服此缺点。在文献 [3] 中给出了另一种方法的详细解释，即使用雅可比转置方法将更新步骤 6 替换为：

$$\mathbf{q} \leftarrow \mathbf{q} + \alpha \mathbf{J}_{eA}^T \Delta \mathbf{x}_e \quad (2.231)$$

如果选择参数 α 足够小，可以保证问题收敛，这个方法的好处是不需要求解耗时的逆。

同样我们也要注意, 参数化的选择导致不同的数值行为, 我们将在实例 2.9.3 中讨论。

【实例 2.9.2】旋转的逆运动学



作为说明不同参数化效果的实例, 我们考虑一个简单机械臂, 该机械臂具有绕 z 、 y 以及 x 轴的三个连续旋转关节, 对应于广义坐标:

$$\mathbf{q} = \begin{pmatrix} q_z \\ q_y \\ q_x \end{pmatrix} \quad (2.232)$$

这组广义坐标的选择导致公式 2.41 中给出的旋转矩阵 $z = q_z$ 、 $y = q_y$ 以及 $x = q_x$ 。代入公式 2.164, 基本的雅可比是:

$$\mathbf{J}_{e0R} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \mathbf{C}_z(q_z) \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \mathbf{C}_z(q_z) \mathbf{C}_y(q_y) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad (2.233)$$

$$= \begin{bmatrix} 0 & -\sin(z) & \cos(y) \cos(z) \\ 0 & \cos(z) & \cos(y) \sin(z) \\ 1 & 0 & -\sin(y) \end{bmatrix} \quad (2.234)$$

初始条件为 $\mathbf{q}^0 = \begin{pmatrix} -0.7 & 0 & 1.5 \end{pmatrix}^T$ 。对于这种位形, 以欧拉 ZYX 坐标表示的末端执行器的旋转为 $\chi_{R \text{ euler ZYX}} = \begin{pmatrix} -0.7 & 0 & 1.5 \end{pmatrix}^T$, 由

以单位球体中心为原点的蓝色坐标系表示。请注意：由于欧拉角参数化与关节旋转的顺序完全相同，所以我们有 $\chi_{R \text{ eulerZYX}} = q$ 。末端执行器的目标方向选择为 $\chi_{R \text{ eulerZYX}} = \begin{pmatrix} 0.7 & 1.5 & -0.5 \end{pmatrix}^T$ (红色框)。对于旋转的参数化，我们有以下选择：

1. 欧拉角 ZYX: $\chi_{R \text{ eulerZYX}} = \begin{pmatrix} z & y & x \end{pmatrix}^T$
2. 欧拉角 XYZ: $\chi_{R \text{ eulerXYZ}} = \begin{pmatrix} x & y & z \end{pmatrix}^T$
3. 旋转向量: $\chi_{\text{rotvec}} = \begin{pmatrix} \varphi_x & \varphi_y & \varphi_z \end{pmatrix}^T$

对于每个参数化方式，必须应用以下步骤：

1. 对于实际的广义坐标 \mathbf{q}^i 评估旋转矩阵 $\mathbf{C}^i = \mathbf{C}(\mathbf{q})$
2. 将 \mathbf{C}^i 变换成末端执行器参数 χ_R^i ，可以使用以下公式：
 - 欧拉 ZYX : 公式 2.42
 - 欧拉 XYZ : 公式 2.45
 - 旋转向量: 公式 2.47、公式 2.49 和公式 2.50
3. 计算解析雅可比 $\mathbf{J}_{eAR}^i = \mathbf{E}_{\chi_R} (\chi_R^i)^{-1} \mathbf{J}_{e0R} (\mathbf{q}^i)$ ，其中 \mathbf{E}_{χ_R} 为：
 - 欧拉 ZYX : 公式 2.86
 - 欧拉 XYZ : 公式 2.88
 - 旋转向量: 公式 2.106
4. 计算差异 $\Delta \chi_R^i = \chi_R^* - \chi_R^i$
5. 更新广义坐标 $\mathbf{q}^{i+1} = \mathbf{q}^i + k \mathbf{J}_{eAR}^+ \Delta \chi_R^i$ ，其中 k 作为一个小的比例因子，保证始终遵循局部线性化。

取决于机器人的运动学和旋转参数化的选择 (蓝色: 欧拉 ZYX, 红色: 欧拉 XYZ, 黑色: 旋转向量)，坐标系以不同的方式旋转到目标系统。此实例的 matlab 代码可在附录 A.2 中找到。

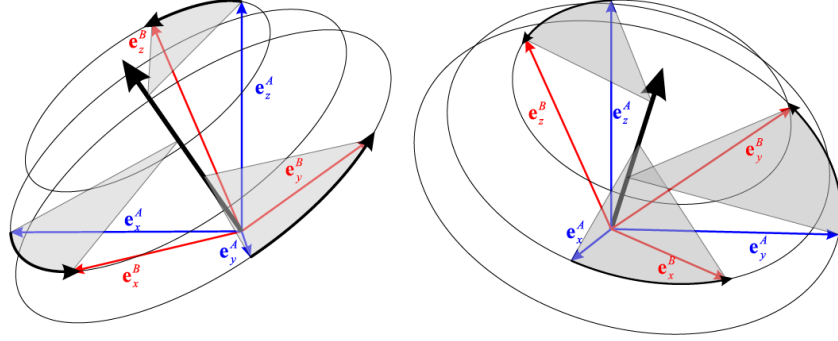


图 2.14: 从起点到目标坐标系的 3D 旋转可以由旋转角度和旋转向量表示

适当的旋转误差

正如我们在前面的例子中看到的，旋转参数化的选择改变了从开始到目标位形的收敛过程。理想情况下，我们将处理允许在旋转“最短路径”上的旋转误差和方向。为了更好地理解三维空间中的旋转误差，我们可以看一下图 2.14。正如我们从第 2.4.5 节中已经知道的，一个可能的旋转参数化方法是轴角，即给出旋转方向的轴（黑色箭头）和定义旋转幅度的角度。由于旋转轴在整个旋转过程中保持恒定，这可以理解为“最近的可能旋转”。因此，我们选择旋转误差为：

$$\Delta\phi = \Delta\varphi \quad (2.235)$$

其参数化从开始 $\mathbf{C}_{SI}(\varphi^t)$ 到达目标方向 $\mathbf{C}_{GI}(\varphi^*)$ 的相对旋转 \mathbf{C}_{GS} ：

$$\mathbf{C}_{GS}(\Delta\varphi) = \mathbf{C}_{GI}(\varphi^*) \mathbf{C}_{SI}^T(\varphi^t) \quad (2.236)$$

请注意 $\Delta\varphi$ 不是 $\varphi^* - \varphi^t$ ，但是可以使用公式 2.47、公式 2.49 和公式 2.50 从 \mathbf{C}_{GS} 得到。给定 2.70 中角速度的定义，我们可以将算法 1 的更新步骤 6 更改为：

$$\mathbf{q} \leftarrow \mathbf{q} + k_{pR} \mathbf{J}_{e0R}^+ \Delta\varphi \quad (2.237)$$

如果 k_{pR} 取值很小，算法将在从开始到目标位形的“最短路径”上旋转。请注意两件事：首先，此更新步骤采用几何而不是解析雅可比。第二，与之前的公式相反， $\Delta\varphi$ 不等于开始和目标旋转的旋转向量的差，而是表示“差异旋转”的旋转向量。

为了更好地理解两坐标系之间相对旋转的概念，我们可以将两坐标系之间的相对旋转与线性相对位置联系起来。考虑以 B 和 B^* 的原点的两个坐标系 \mathcal{B}

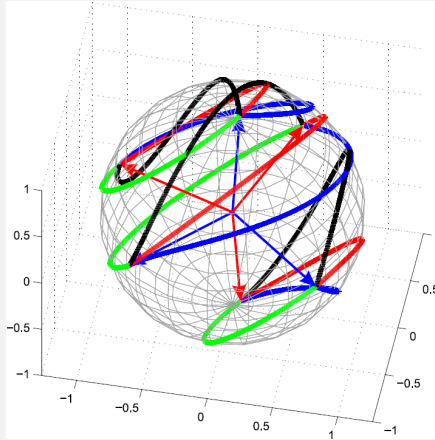
和 \mathcal{B}^* 和，他们表示刚体相对于惯性坐标系 \mathcal{I} 的测量和期望姿态。线性位置误差可以定义为：

$$\mathcal{I}\mathbf{r}_{BB^*} = \mathcal{I}\mathbf{r}_{IB^*} - \mathcal{I}\mathbf{r}_{IB} \quad (2.238)$$

类似地，我们可以将两个姿态之间的相对旋转定义为：

$$\mathbf{C}_{BB^*} = \mathbf{C}_{IB}^T \cdot \mathbf{C}_{IB^*} \quad (2.239)$$

【实例 2.9.3】使用旋转向量的逆运动学



本实例考虑了与实例 2.9.3 中相同的问题，但是使用旋转向量描述误差的方式来实现逆运动学方法。

给定目标旋转矩阵 \mathbf{C}^* ，应用以下步骤：

1. 相对于实际的广义坐标 \mathbf{q}^i 获取旋转矩阵 $\mathbf{C}^i = \mathbf{C}(\mathbf{q})$
2. 计算相对旋转 $\mathbf{C}^{rel} = \mathbf{C}^* \cdot \mathbf{C}^{iT}$
3. 使用公式 2.47、公式 2.49 和公式 2.50 计算相对旋转的旋转向量 $\Delta\varphi$ 。
4. 更新广义坐标 $q^{i+1} = q^i + k J_{e_{0R}}^+ \Delta\varphi$ ，其中 k 作为一个小的比例因子，始终遵循局部线性化。

如我们所见，解遵循从开始到目标位置的绿色路径 (= “直线”)。此实例的 matlab 代码可在附录 A.2 中找到。

2.9.4 轨迹控制

纯逆微分运动学通常不足以跟随预定义的任务空间轨迹，因为如果没有为位置或旋转提供反馈，姿态将会漂移。因此，轨迹控制可以通过利用加权跟踪误差来反馈稳定逆微分运动学，最终会得到平滑并无漂移的运动轨迹。

位置轨迹控制

具有给定预先计划位置的位置跟踪 $\mathbf{r}_e^*(t)$ 和速度 $\dot{\mathbf{r}}_e^*(t)$ 问题是非常简单的。反馈控制部分保证误差：

$$\Delta \mathbf{r}_e^t = \mathbf{r}_e^*(t) - \mathbf{r}_e(\mathbf{q}^t) \quad (2.240)$$

为零，这将得到以下轨迹控制器：

$$\dot{\mathbf{q}}^* = \mathbf{J}_{e0_P}^+(\mathbf{q}^t) \cdot (\dot{\mathbf{r}}_e^*(t) + k_{p_P} \Delta \mathbf{r}_e^t) \quad (2.241)$$

因此 k_{p_P} 表示位置反馈，定义了实际位置收敛到目标位置的速度。如果该算法实现为具有固定时间步长 Δt 的数字控制问题，可以通过特征值分析来稳定 k_{p_P} 边界（见例如 [8]）。

方向轨迹控制

跟踪方向 $\chi_R^*(t)$ 和角速度 $\omega^*(t)$ 时，问题将会变得越来越复杂。正如我们在逆运动学实例 2.9.3 中看到的，参数化的选择导致了不同的轨迹。最好的方法是遵循最短旋转方法，这相当于位置跟踪。有了这个，我们可以写出：

$$\dot{\mathbf{q}}^* = \mathbf{J}_{e0_R}^+(\mathbf{q}^t) \cdot (\omega_e^*(t) + k_{p_R} \Delta \varphi) \quad (2.242)$$

即 $\Delta \varphi$ 按照第 2.9.3 小节所述进行计算。请注意，该公式不需要解析雅可比。

2.10 浮动基座运动学

图 2.15 中描述的四足和人形的浮动基座机器人可以由 n_b 个未驱动的基本坐标 \mathbf{q}_b 和 n_j 个驱动关节坐标 \mathbf{q}_j 描述：

$$\mathbf{q} = \begin{pmatrix} \mathbf{q}_b \\ \mathbf{q}_j \end{pmatrix} \quad (2.243)$$

未驱动的基座在平移和旋转上都是自由的：

$$\mathbf{q}_b = \begin{pmatrix} \mathbf{q}_{b_P} \\ \mathbf{q}_{b_R} \end{pmatrix} \in \mathbb{R}^3 \times SO(3) \quad (2.244)$$

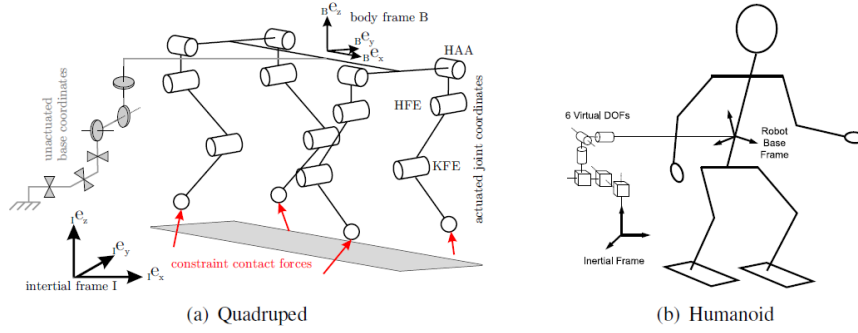


图 2.15: 自由浮动机器人的基座未驱动

其中位置 \mathbf{q}_{bP} 和旋转 \mathbf{q}_{bR} 可以使用不同的表示来参数化, 如第 2.2.1 小节和 2.4.5 小节所述, 因此, 浮基系统的广义坐标向量的维数为 $n_b + n_j$, 但取决于旋转的参数化形式。此时, 表示基座的广义坐标的最小数量为 $n_{b0} = 6$ 。

浮基系统有几个挑战。首先, 通常没有 (机载) 传感器可以直接测量基座的位置和方向。为了解决这个问题, 在一些研究领域中研究人员会使用运动跟踪系统, 即利用的外部摄像机来跟踪和测量附着在机器人上的标记点, 从而测量基座的姿势。否则, 有必要使用传感器融合算法来整合来自不同传感器的信息从而估计姿态。第二个挑战是, 由于基座不是直接驱动的, 所以刚体系统的运动 (比如说总线性和角动量) 只能通过接触产生的额外的外力来改变 (另见第 2.10.4 小节)。

2.10.1 广义速度和加速度

因为 $SO(3)$ 的微分不同于 \mathbb{R}^3 , 所以人们经常引入广义速度和加速度向量:

$$\mathbf{u} = \begin{pmatrix} \mathcal{T}\mathbf{v}_B \\ \mathcal{B}\boldsymbol{\omega}_{IB} \\ \dot{\varphi}_1 \\ \vdots \\ \dot{\varphi}_{n_j} \end{pmatrix} \in \mathbb{R}^{6+n_j} = \mathbb{R}^{n_u} \quad \dot{\mathbf{u}} = \begin{pmatrix} \mathcal{T}\mathbf{a}_B \\ \mathcal{B}\dot{\boldsymbol{\phi}}_{IB} \\ \ddot{\varphi}_1 \\ \vdots \\ \ddot{\varphi}_{n_j} \end{pmatrix} \in \mathbb{R}^{6+n_j} \quad (2.245)$$

如第 2.5.1 节所述, 存在从旋转速度 $\boldsymbol{\omega}$ 到末端执行器旋转坐标 \mathbf{q}_{bR} 相对于

时间的导数的直接映射，使得：

$$\mathbf{u} = \mathbf{E}_{fb} \cdot \dot{\mathbf{q}} \quad \text{其中} \quad \mathbf{E}_{fb} = \begin{bmatrix} \mathbb{I}_{3 \times 3} & 0 & 0 \\ 0 & \mathbf{E}_{\mathbf{x}_R} & 0 \\ 0 & 0 & \mathbb{I}_{n_j \times n_j} \end{bmatrix} \quad (2.246)$$

其中旋转参数化相关矩阵 $\mathbf{E}_{\mathbf{x}_R}$ 以及它的逆 $\mathbf{E}_{\mathbf{x}_R}^{-1}$ 由以下参数化形式给出：

- 欧拉角
 - XYZ : 公式 2.88 和公式 2.89
 - ZYX : 公式 2.86 和公式 2.87
 - ZYZ : 公式 2.90 和公式 2.91
- 四元数公式 2.98 和公式 2.99
- 轴角公式 2.102 和公式 2.103
- 旋转向量公式 2.106 和公式 2.107

请注意，许多教科书都使用 $\dot{\mathbf{q}}$ ，但隐式代表的是 \mathbf{u} ，而不是参数化的时间导数。对于固定基座系统来说，上述两个符号表示通常没有区别。但是，无论何时你在使用浮动基座系统时，都必须意识到其中的区别。在这些讲义中，我们试图在这种情况下保持一致，因此当具体谈论浮动基础系统时，我们总是使用 \mathbf{u} 。

2.10.2 正向运动学

我们希望导出广义速度 \mathbf{u} 和操作空间中点 Q 的速度 ${}^I\mathbf{v}_Q$ 之间的关系，点 Q 固定在以浮动基座 B 为起点的运动链的末端。点 Q 对应的位置向量 ${}^I\mathbf{r}_{IQ} = {}^I\mathbf{r}_{IQ}(\mathbf{q})$ 在惯性坐标系 \mathcal{I} 中的表示由以下公式给出：

$${}^I\mathbf{r}_{IQ}(\mathbf{q}) = {}^I\mathbf{r}_{IB}(\mathbf{q}) + \mathbf{C}_{IB}(\mathbf{q}) \cdot {}^B\mathbf{r}_{BQ}(\mathbf{q}) \quad (2.247)$$

其中旋转矩阵 $\mathbf{C}_{IB}(\mathbf{q})$ 描述浮动底座 B 相对于惯性坐标系 \mathcal{I} 的方向。 ${}^I\mathbf{r}_{IB}(\mathbf{q})$ 表示在惯性坐标系中浮基坐标系 B 相对于惯性坐标系 \mathcal{I} 的位置， ${}^B\mathbf{r}_{BQ}(\mathbf{q})$ 代表在惯性坐标系 B 中点 Q 相对于浮基 B 的位置，以及 $\mathbf{q} = \mathbf{q}(t)$ 是时间 t 的函数。

2.10.3 浮动基座系统的微分运动学

根据公式 2.247，位置向量的时间微分可以由以下公式得到：

$${}^I\mathbf{v}_Q = {}^I\mathbf{v}_B + \dot{\mathbf{C}}_{IB} \cdot {}^B\mathbf{r}_{BQ} + \mathbf{C}_{IB} \cdot {}^B\dot{\mathbf{r}}_{BQ}$$

$$\begin{aligned}
&= {}^I\mathbf{v}_B + \mathbf{C}_{IB} \cdot [{}^B\boldsymbol{\omega}_{IB}]_{\times} \cdot {}^B\mathbf{r}_{BQ} + \mathbf{C}_{IB} \cdot {}^B\dot{\mathbf{r}}_{BQ} \\
&= {}^I\mathbf{v}_B - \mathbf{C}_{IB} \cdot [{}^B\mathbf{r}_{BQ}]_{\times} \cdot {}^B\boldsymbol{\omega}_{IB} + \mathbf{C}_{IB} \cdot {}^B\dot{\mathbf{r}}_{BQ} \\
&= {}^I\mathbf{v}_B - \mathbf{C}_{IB} \cdot [{}^B\mathbf{r}_{BQ}]_{\times} \cdot {}^B\boldsymbol{\omega}_{IB} + \mathbf{C}_{IB} \cdot {}^B\mathbf{J}_{P_{q_j}}(\mathbf{q}_j) \cdot \dot{\mathbf{q}}_j \\
&= \begin{bmatrix} \mathbb{I}_{3 \times 3} & -\mathbf{C}_{IB} \cdot [{}^B\mathbf{r}_{BQ}]_{\times} & \mathbf{C}_{IB} \cdot {}^B\mathbf{J}_{P_{q_j}}(\mathbf{q}_j) \end{bmatrix} \cdot \mathbf{u} \quad (2.248)
\end{aligned}$$

如果我们在 ${}^I\mathbf{r}_{IQ}$ 表示的点处附加一个坐标系, 我们可以导出角速度的类似映射。坐标系 Q 相对于惯性坐标系 I 的方向描述如下:

$$\mathbf{C}_{IQ} = \mathbf{C}_{IB} \cdot \mathbf{C}_{BQ} \quad (2.249)$$

对公式 2.249 等式两边关于时间求导得到:

$$\begin{aligned}
[{}^I\boldsymbol{\omega}_{IQ}]_{\times} \cdot \mathbf{C}_{IQ} &= [{}^I\boldsymbol{\omega}_{IB}]_{\times} \cdot \mathbf{C}_{IB} \cdot \mathbf{C}_{BQ} + \mathbf{C}_{IB} \cdot [{}^B\boldsymbol{\omega}_{BQ}]_{\times} \cdot \mathbf{C}_{BQ} \\
&= [{}^I\boldsymbol{\omega}_{IB}]_{\times} \cdot \mathbf{C}_{IQ} + \mathbf{C}_{IB} \cdot \mathbf{C}_{BI} \cdot [{}^I\boldsymbol{\omega}_{BQ}]_{\times} \cdot \mathbf{C}_{BI}^T \cdot \mathbf{C}_{BQ} \\
&= [{}^I\boldsymbol{\omega}_{IB}]_{\times} \cdot \mathbf{C}_{IQ} + [{}^I\boldsymbol{\omega}_{BQ}]_{\times} \cdot \mathbf{C}_{IQ} \quad (2.250)
\end{aligned}$$

最后给出:

$${}^I\boldsymbol{\omega}_{IQ} = {}^I\boldsymbol{\omega}_{IB} + {}^I\boldsymbol{\omega}_{BQ} \quad (2.251)$$

$$= \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{C}_{IB} & \mathbf{C}_{IB} \cdot {}^B\mathbf{J}_{R_{q_j}}(\mathbf{q}_j) \end{bmatrix} \cdot \mathbf{u} \quad (2.252)$$

因此, 从广义速度 \mathbf{u} 到相对于坐标系 Q 的操作空间旋量 $\begin{bmatrix} {}^I\mathbf{v}_Q^T & {}^I\boldsymbol{\omega}_{IQ}^T \end{bmatrix}^T$ 的映射可以由空间雅可比实现:

$${}^I\mathbf{J}_Q(\mathbf{q}) = \begin{bmatrix} {}^I\mathbf{J}_P \\ {}^I\mathbf{J}_R \end{bmatrix} = \begin{bmatrix} \mathbb{I}_{3 \times 3} & -\mathbf{C}_{IB} \cdot [{}^B\mathbf{r}_{BQ}]_{\times} & \mathbf{C}_{IB} \cdot {}^B\mathbf{J}_{P_{q_j}}(\mathbf{q}_j) \\ \mathbf{0}_{3 \times 3} & \mathbf{C}_{IB} & \mathbf{C}_{IB} \cdot {}^B\mathbf{J}_{R_{q_j}}(\mathbf{q}_j) \end{bmatrix} \quad (2.253)$$

2.10.4 接触和约束

在运动学中, 机器人与其环境之间的接触可以被建模为运动学约束。在坐标系 I 中, 与环境接触的每一点 C_i 施加的三个约束表示为:

$${}^I\mathbf{r}_{IC_i} = \text{const}, \quad {}^I\dot{\mathbf{r}}_{IC_i} = {}^I\ddot{\mathbf{r}}_{IC_i} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (2.254)$$

这些接触约束可以用接触点雅可比表示为广义速度和加速度的函数:

$${}^I\mathbf{J}_{C_i}\mathbf{u} = 0 \quad {}^I\mathbf{J}_{C_i}\dot{\mathbf{u}} + {}^I\dot{\mathbf{J}}_{C_i}\mathbf{u} = 0 \quad (2.255)$$

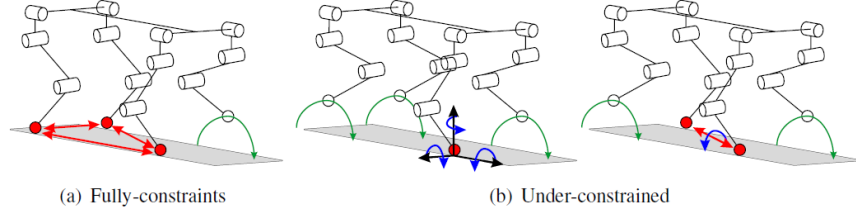


图 2.16: 根据点接触的数量和布置不同, 四足机器人可能处于完全约束 (a) 或受欠约束 (b) 状态

如果有 n_c 个活动的接触点, 约束简单地堆叠得到:

$$\mathbf{J}_c = \begin{bmatrix} \mathbf{J}_{C_1} \\ \vdots \\ \mathbf{J}_{C_{n_c}} \end{bmatrix} \in \mathbb{R}^{3n_c \times n_n} \quad (2.256)$$

约束雅可比矩阵的秩 $\text{rank}(\mathbf{J}_c)$ 表示独立接触约束的数量。这个堆叠的雅可比可以被分成:

$$\mathbf{J}_c = \begin{bmatrix} \mathbf{J}_{c,b} & \mathbf{J}_{c,j} \end{bmatrix} \quad (2.257)$$

其中 $\mathbf{J}_{c,b}$ 表示基座运动和接触约束之间的关系。如果 $\mathbf{J}_{c,b}$ 满秩即 $\text{rank}(\mathbf{J}_{c,b}) = 6$, 也就意味着该系统具有足够的约束, 使得基座的运动可以由关节运动控制。独立接触约束数量 $\text{rank}(\mathbf{J}_c)$ 与基本约束的数量 $\text{rank}(\mathbf{J}_{c,b})$ 之间的差是必须满足的内部运动学约束的数量。

2.10.5 点触点-四足

四足机器人的每一个点状脚都能施加三个独立约束。在两点接触的情况下, 堆叠接触雅可比的秩 $\text{rank}(\mathbf{J}_c) = 6$, 但是相对于基坐标系的雅可比的秩只有 $\text{rank}(\mathbf{J}_{c,b}) = 5$ 。这意味着系统欠驱动, 底座不能被接关节任意移动。当观察图 2.16(b) 时, 这变得直观清晰, 因为机器人不能改变支撑线周围的方向。

与此相反的是, 如图 2.16(a) 所示的三点接触意味着 $\text{rank}(\mathbf{J}_c) = 9$ 和 $\text{rank}(\mathbf{J}_{c,b}) = 6$ 。这意味着刚体的位置和方向通过关节驱动是完全可控的。同时, 有三个内部约束可以通过三个腿不能相对于另一个移动这一事实来解释。

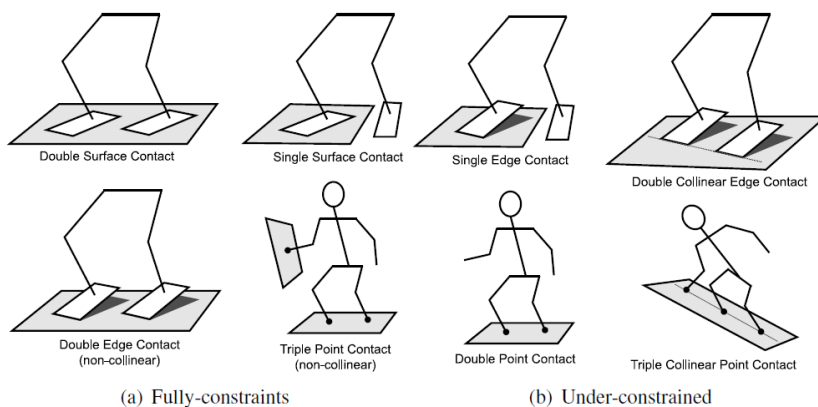


图 2.17: 根据点接触的数量和分布不同, 人形机器人是完全受约束 (a) 或欠约束 (b) 状态

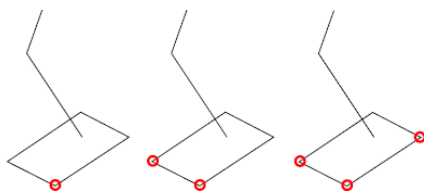


图 2.18: 单脚有多个接触点

2.10.6 扩展接触-人形

对于有平面脚的系统的来说, 需要额外的约束以限制脚的旋转。一个可能的选择是引入旋转雅可比。更常见的是在同一连杆上分配多个接触点。单个接触点 (图 2.18, 左) 施加了三个约束, 正如我们上一小节中已经看到的。在两个接触点的情况下, 约束的秩为 $\text{rank}(\mathbf{J}_c) = \text{rank}(\mathbf{J}_{c,b}) = 5$ 。如果将三个点分配给同一元素, 虽然 $\mathbf{J}_c \in \mathbb{R}^{9 \times n_j}$, 我们将得到 $\text{rank}(\mathbf{J}_c) = \text{rank}(\mathbf{J}_{c,b}) = 6$ 。

2.10.7 支持一致逆运动学

将逆运动学应用于浮动基座系统让我们能够在不违反接触约束的情况下移动单个关节以实现某些任务空间运动。具有优先次序的多任务方法也可以应用这个方法, 此时接触约束被认为比任务空间运动具有更高的优先级。如之前所

述, n_c 条腿与地面接触时的接触约束为:

$$\mathbf{J}_c \mathbf{u} = \mathbf{0} \quad (2.258)$$

这意味着系统在与地面接触时的运动为:

$$\mathbf{u} = \mathbf{J}_c^+ \mathbf{0} + \mathcal{N}(\mathbf{J}_c) \mathbf{u}_0 = \mathbf{N}_c \mathbf{u}_0 \quad (2.259)$$

因此, 给定一个需要的任务空间运动:

$$\mathbf{w}_t = \mathbf{J}_t \mathbf{u}_i \quad (2.260)$$

达到此目的所需的关节速度是:

$$\mathbf{u} = \mathbf{N}_c (\mathbf{J}_t \mathbf{N}_c)^+ + \mathbf{w}_t \quad (2.261)$$

第三章 动力学

3.1 介绍

对于大多数固定基座机器人来说，我们能够找到一个通用的多体动力学方程来描述系统的动力学，该方程为：

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{J}_c(\mathbf{q})^T \mathbf{F}_c \quad (3.1)$$

多体动力学方程由以下项组成：

$\mathbf{M}(\mathbf{q})$	$\in \mathbb{R}^{n_q \times n_q}$	正交广义质量矩阵
$\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$	$\in \mathbb{R}^{n_q}$	广义位置、速度和加速度向量
$\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})$	$\in \mathbb{R}^{n_q}$	科式力与离心力
$\mathbf{g}(\mathbf{q})$	$\in \mathbb{R}^{n_q}$	重力项
$\boldsymbol{\tau}$	$\in \mathbb{R}^{n_q}$	外部广义力
\mathbf{F}_c	$\in \mathbb{R}^{3 \times n_c}$	外部笛卡尔力
$\mathbf{J}_c(\mathbf{q})$	$\in \mathbb{R}^{n_c \times n_q}$	对应于外部力的几何雅可比

请注意：为了行文的简单和紧凑，我们在这里使用固定基座系统中使用的变量 $\ddot{\mathbf{q}}$ 而不是浮动基座系统引入的变量 $\ddot{\mathbf{u}}$ 。我们将在第 3.7 节具体论述浮动基座系统的动力学。

在不同的文献中存在不同的方法来计算给定系统的运动方程，运动方程也叫封闭形式的系统动力学数学模型。几乎所有方法都是基于牛顿或者拉格朗日力学，这两种方法得到的动力学描述都是等效的。

在本文中，我们将介绍机器人学科中最常用的几种方法。在第 3.3 节中，我们将介绍第一种方法，也就是著名的经典牛顿-欧拉方法，它在机器人所有连杆上应用线性动量和角动量守恒原理，并明确考虑笛卡尔空间中的运动。在第 3.4 节中，我们将提出第二种方法，这种方法被称为拉格朗日方法，是一种在广义坐标空间上根据最小作用量原理，利用标量能量函数来对系统进行分析的技术，

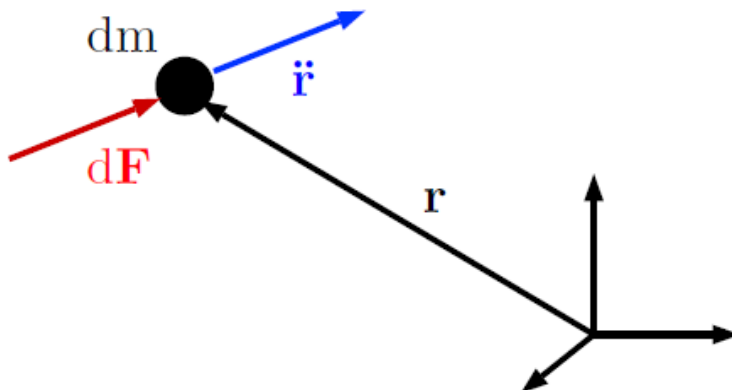


图 3.1: 作用于单个粒子的力

这种方法得到的解将自动满足系统运动学约束。第三种方法是牛顿-欧拉投影法，可以说是最有用的方法，将在第 3.5 节中介绍，该方法结合了牛顿-欧拉法和拉格朗日法的优点，在广义坐标基础上重构了牛顿-欧拉方法，因此可以直接得到系统的可行运动，即满足运动学约束的运动。

尽管这些方法通常被认为是完全不同的，并且初看他们确实存在很大的差异性，但它们的等价性在你深入了解后将变得非常清晰。最后，由读者自己来确定机器人系统建模和问题构造所采用的方法。

3.2 经典力学基础

在介绍导出运动方程的上述方法之前，我们必须首先熟悉前置的原理和必要的分析工具，这些是我们理解后文将要介绍的方法的基础。

3.2.1 牛顿粒子定律

牛顿力学用最简单的形式描述了质点的运动，质点是拥有质量 m 的并且不存在体积或形状的点，是物理学中的一个理想化模型。我们可以假设质点的质量全部集中在由位置向量 \mathbf{r} 定义的点处，如图 3.1 所示。请记住，质点没有方向，因为质点不具有任何几何形状，这意味着我们只需要利用牛顿第二定律就可以完整地描述系统的运动：

$$\ddot{\mathbf{r}}m = \mathbf{F} \quad (3.2)$$

再次考虑到我们已经将粒子的质量定义在无限小的维度这一事实，我们可以将其视为无穷小的质量 dm 在粒子的位置上集中受到无穷小的力 $d\mathbf{F}$ ：

$$\ddot{\mathbf{r}}dm = d\mathbf{F} \quad (3.3)$$

正如我们将在接下来的章节中看到的，这些关系在由质点衍生出来的场景中仍然发挥着重要作用。公式 3.2 同样适用于刚体质心的线性动力学的计算，而公式 3.3 则通过虚功原理展现了系统中内力所带来的影响。

3.2.2 虚位移

我们将通过操作符 δ 来介绍变分符号概念，从本质上来说，变分操作符 δ 与微分运算符 d 除了含义不同之外，其他行为完全相同。微分被用来描述无穷小量，紧接着引申出了一个量相对于另一个量的“变化率”。相反的是，变分描述了在一个固定的时间瞬间，物理量在遵守约束的前提下可能位于的所有可能方向。

关于变分，我们要记住的最重要的事实是：如果它是关于时间的函数，在处理变分时，我们可以将时间完全忽略。为了证明这一性质，我们定义位置向量的容许变化为虚位移 $\delta\mathbf{r}$ ，并假设位置是广义坐标 \mathbf{q} 和时间 t 的函数，因此可以得到： $\mathbf{r} = \mathbf{r}(\mathbf{q}, t)$ 。通过对 \mathbf{q} 的 n_q 个元素应用链式法则，我们可以计算变分 $\delta\mathbf{r}$ ：

$$\delta\mathbf{r}(\mathbf{q}, t) = \sum_{k=1}^{n_q} \frac{\partial \mathbf{r}}{\partial q_k} \delta q_k \quad (3.4)$$

请注意，公式 3.4 中的表达式不包括 $\frac{\partial \mathbf{r}}{\partial t}$ ，因此此时的变分可以被理解为只用几何关系就可以解释的位移。

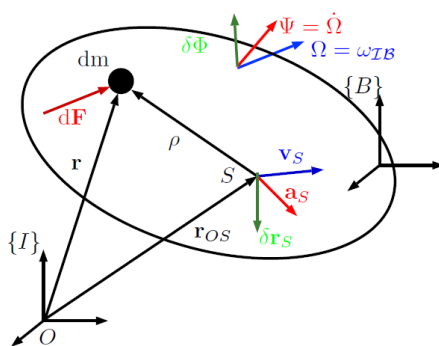


图 3.2: 单刚体运动学

3.2.3 单个刚性体的虚位移

如图 3.2 所示, 在 3D 笛卡尔空间中存在有质量的物体, 该物体可以被等价成被紧密放置在一起大量的粒子形成的单个刚体。刚体上每个无穷小的质点 dm 受制于全身的运动, 因此可以在任何时刻被指定一个绝对位置和速度。现在让我们考虑刚体上另一点 S , 这样我们就可以定义 dm 相对于点 S 的相对位置 ρ 。

因此, 可以使用第 2.7 节中介绍的刚体运动学公式, 通过另一个位于刚体上的点 S 来描述任意定义在刚体 B 上的质点 dm 的运动:

$$\mathbf{r} = \mathbf{r}_{OS} + \rho \quad (3.5)$$

$$\dot{\mathbf{r}} = \mathbf{v}_S + \boldsymbol{\Omega} \times \rho = \begin{bmatrix} \mathbb{I}_{3 \times 3} & -[\rho]_{\times} \end{bmatrix} \begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\Omega} \end{pmatrix} \quad (3.6)$$

$$\ddot{\mathbf{r}} = \mathbf{a}_S + \boldsymbol{\Psi} \times \rho + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \rho) = \begin{bmatrix} \mathbb{I}_{3 \times 3} & -[\rho]_{\times} \end{bmatrix} \begin{pmatrix} \mathbf{a}_s \\ \boldsymbol{\Psi} \end{pmatrix} + [\boldsymbol{\Omega}]_{\times} [\boldsymbol{\Omega}]_{\times} \rho \quad (3.7)$$

其中 \mathbf{r}_{OS} 是刚体上点 S 的绝对位置, ρ 是 dm 相对于 S 的相对位置, 以及 \mathbf{v}_S 和 $\boldsymbol{\Omega}$ 表示点 S 的绝对速度, $\boldsymbol{\Psi}$ 和 \mathbf{a}_S 表示点 S 的绝对加速度。将公式 3.4 代入公式 3.5, 我们得到了刚体元素 dm 的虚位移的表达式:

$$\delta \mathbf{r} = \delta \mathbf{r}_S + \delta \boldsymbol{\Phi} \times \rho = \begin{bmatrix} \mathbb{I}_{3 \times 3} & -[\rho]_{\times} \end{bmatrix} \begin{pmatrix} \delta \mathbf{r}_s \\ \delta \boldsymbol{\Phi} \end{pmatrix} \quad (3.8)$$

其中 $\delta \boldsymbol{\Phi}$ 表示固定在刚体上 S 点的局部坐标系的微小旋转的变分, 其描述了位于 S 点处的局部坐标系在满足约束的前提下方向的变化。注意 $\delta \boldsymbol{\Phi}$ 都是粗体表示, 这是因为变分不是相对于方向的数值求解, 就像角速度也不是方向在时间上求微分的结果一样。

3.2.4 多体系统的虚位移

多体系统只能实现满足关节约束的运动, 从而限制了连杆之间的相对运动。正如我们已经在 2.8 节看到的, 运动通常使用广义坐标 \mathbf{q} 来描述:

$$\begin{pmatrix} \mathbf{v}_s \\ \boldsymbol{\Omega} \end{pmatrix} = \begin{bmatrix} \mathbf{J}_P \\ \mathbf{J}_R \end{bmatrix} \dot{\mathbf{q}} \quad (3.9)$$

$$\begin{pmatrix} \mathbf{a}_s \\ \boldsymbol{\Psi} \end{pmatrix} = \begin{bmatrix} \mathbf{J}_P \\ \mathbf{J}_R \end{bmatrix} \ddot{\mathbf{q}} + \begin{bmatrix} \dot{\mathbf{J}}_P \\ \dot{\mathbf{J}}_R \end{bmatrix} \dot{\mathbf{q}} \quad (3.10)$$

再次将公式 3.4 代入到公式 3.9 和公式 3.10 中，虚位移的概念现在引入到多体系统中，并且意味着满足关节约束的虚位移必须具有以下形式：

$$\begin{pmatrix} \delta \mathbf{r}_s \\ \delta \Phi \end{pmatrix} = \begin{bmatrix} \mathbf{J}_P \\ \mathbf{J}_R \end{bmatrix} \delta \mathbf{q} \quad (3.11)$$

3.2.5 虚功原理

力学中的一个基本原理是虚功原理，它描述了这样一个事实：即位形约束实际上定义了沿虚位移方向不做功的力。对于任意关节产生的作用于 \mathbf{r}_c 点的约束力 \mathbf{F}_c ，虚功原理告诉我们：

$$\delta W = \delta \mathbf{r}_c^T \cdot \mathbf{F}_c = 0 \quad (3.12)$$

我们可以通过考虑在刚体 \mathcal{B} 里的一个无限小的质点 dm 的基础上来做进一步扩展，考虑多个粒子之间的关系。通过引入描述粒子动力学平衡的达朗贝尔原理，我们可以得到整个物体 \mathcal{B} 在沿几何形状上的虚功的积分：

$$\delta W = \int_{\mathcal{B}} \delta \mathbf{r}^T \cdot (\ddot{\mathbf{r}} dm - d\mathbf{F}_{ext}) = 0, \quad \forall \delta \mathbf{r} \quad (3.13)$$

其中各项定义如下：

dm	无限小质量单元
$d\mathbf{F}_{ext}$	施加到单元 dm 的外力
\mathbf{r}	单元 dm 的加速度
$\delta \mathbf{r}$	单元 dm 的虚位移
\mathcal{B}	包含无穷小质量 dm 的粒子的子刚体系统

因此，对于所有满足系统约束的位形变化 $\delta \mathbf{r}$ 来说，虚功都为零。尽管这个表述可能看起来有些复杂，但在后续的讲义中会变得更加清楚，它以一种非常紧凑和易于理解的方式描述了我们力学中大多数众所周知的概念。

3.3 牛顿-欧拉法

3.3.1 单体牛顿欧拉

在单刚体上应用虚功原理公式 3.13 可得到：

$$0 = \delta W = \int_{\mathcal{B}} \begin{pmatrix} \delta \mathbf{r}_s \\ \delta \Phi \end{pmatrix}^T \begin{bmatrix} \mathbb{I}_{3 \times 3} \\ [\rho]_{\times} \end{bmatrix} \left(\begin{bmatrix} \mathbb{I}_{3 \times 3} & -[\rho]_{\times} \end{bmatrix} \begin{pmatrix} \mathbf{a}_s \\ \Psi \end{pmatrix} dm + [\Omega]_{\times}^2 \rho dm - d\mathbf{F}_{ext} \right)$$

$$= \begin{pmatrix} \delta \mathbf{r}_s \\ \delta \Phi \end{pmatrix}^T \int_B \left(\begin{bmatrix} \mathbb{I}_{3 \times 3} dm & [\boldsymbol{\rho}]_{\times}^T dm \\ [\boldsymbol{\rho}]_{\times} dm & -[\boldsymbol{\rho}]_{\times}^2 dm \end{bmatrix} \begin{pmatrix} \mathbf{a}_s \\ \Psi \end{pmatrix} + \begin{pmatrix} [\boldsymbol{\Omega}]_{\times}^2 \boldsymbol{\rho} dm \\ [\boldsymbol{\rho}]_{\times} [\boldsymbol{\Omega}]_{\times}^2 \boldsymbol{\rho} dm \end{pmatrix} - \begin{pmatrix} d\mathbf{F}_{\text{ext}} \\ [\boldsymbol{\rho}]_{\times} d\mathbf{F}_{\text{ext}} \end{pmatrix} \right) \quad (3.14)$$

请注意，该形式必须适用于任意虚位移，因为没有关节或接触引入主动约束。已知公式 $\mathbf{a} \times (\mathbf{b} \times (\mathbf{b} \times \mathbf{a})) = -\mathbf{b} \times (\mathbf{a} \times (\mathbf{a} \times \mathbf{b}))$ ，然后引入：

$$\int_B dm =: m \quad \text{刚体质量} \quad (3.15)$$

$$\int_B \boldsymbol{\rho} dm =: \mathbf{0} \quad \text{因为S为质心} \quad (3.16)$$

$$\int_B -[\boldsymbol{\rho}]_{\times}^2 dm = \int_B [\boldsymbol{\rho}]_{\times} [\boldsymbol{\rho}]_{\times}^T dm =: \boldsymbol{\Theta}_S \quad \text{绕质心的惯量} \quad (3.17)$$

我们得到：

$$0 = \delta W = \begin{pmatrix} \delta \mathbf{r}_s \\ \delta \Phi \end{pmatrix}^T \left(\begin{bmatrix} \mathbb{I}_{3 \times 3} m & 0 \\ 0 & \boldsymbol{\Theta}_S \end{bmatrix} \begin{pmatrix} \mathbf{a}_s \\ \Psi \end{pmatrix} + \begin{pmatrix} 0 \\ [\boldsymbol{\Omega}]_{\times} \boldsymbol{\Theta}_S \boldsymbol{\Omega} \end{pmatrix} - \begin{pmatrix} \mathbf{F}_{\text{ext}} \\ \mathbf{T}_{\text{ext}} \end{pmatrix} \right), \forall \begin{pmatrix} \delta \mathbf{r}_s \\ \delta \Phi \end{pmatrix} \quad (3.18)$$

为了定义线动量和角动量守恒定律，我们引入了定义：

$$\mathbf{p}_S = m \mathbf{v}_S \quad \text{线动量} \quad (3.19)$$

$$\mathbf{N}_S = \boldsymbol{\Theta}_S \cdot \boldsymbol{\Omega} \quad \text{绕质心的角动量} \quad (3.20)$$

$$\dot{\mathbf{p}}_S = m \mathbf{a}_S \quad \text{线动量导数} \quad (3.21)$$

$$\dot{\mathbf{N}}_S = \boldsymbol{\Theta}_S \cdot \Psi + \boldsymbol{\Omega} \times \boldsymbol{\Theta}_S \cdot \boldsymbol{\Omega} \quad \text{角动量导数} \quad (3.22)$$

一个自由移动的物体满足线动量的变化等于所有外力的总和：

$$0 = \delta W = \begin{pmatrix} \delta \mathbf{r}_s \\ \delta \Phi \end{pmatrix}^T \left(\begin{pmatrix} \dot{\mathbf{p}}_S \\ \dot{\mathbf{N}}_S \end{pmatrix} - \begin{pmatrix} \mathbf{F}_{\text{ext}} \\ \mathbf{T}_{\text{ext}} \end{pmatrix} \right) \quad \forall \begin{pmatrix} \delta \mathbf{r}_s \\ \delta \Phi \end{pmatrix} \quad (3.23)$$

这引出了著名牛顿-欧拉的公式：

$$\dot{\mathbf{p}}_S = \mathbf{F}_{\text{ext},S} \quad (3.24)$$

$$\dot{\mathbf{N}}_S = \mathbf{T}_{\text{ext}} \quad (3.25)$$

其中 $\mathbf{F}_{\text{ext},S}$ 是通过质心 (COG) 的合外力， \mathbf{T}_{ext} 是合外扭矩。作用线不通过质心作用的外力需要转换到通过质心的等效力和力矩对。请再次注意，对于带数值的计算，线动量和角动量导数项必须在相同的坐标系中表示。对于惯性张量 $\boldsymbol{\Theta}$ 我们必须满足 ${}_B \boldsymbol{\Theta} = \mathbf{C}_{BA} \cdot {}_A \boldsymbol{\Theta} \cdot \mathbf{C}_{BA}^T$ 。

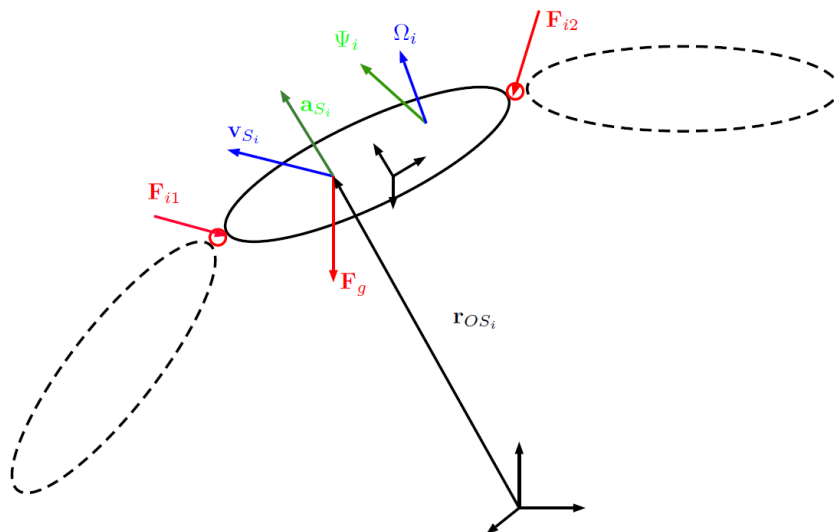


图 3.3: 受到关约束力的多体系统的自由切割

3.3.2 多体系统牛顿欧拉方程

在处理多体系统时，一个有效的方法是如图 3.3 所示在关节处将所有单刚体分割开，并将每个单刚体视为一个单元。因此，约束力 \mathbf{F}_i 变成作用在每个刚体上的外力。对于受到包括关节力 \mathbf{F}_i 在内的外力的所有这些物体，我们需要在所有自由度中应用线性动量守恒公式 3.24 和角动量守恒公式 3.25。对于一般的固定基座三维机器人来说，这对应于 $6n_j$ 个方程组，其中 $5n_j$ 个方程是理想关节的运动限制引入的，它们确保两个相连的物体仅沿着关节的方向移动，而不会在被关节阻碍的其他所有方向上移动。

许多仿真系统建立在这种形式之上，并使用硬约束或软约束来实现关节运动约束。由于可以应用的求解器类型较多，后一种方法即软约束通常会加快仿真速度。然而，它仅近似地满足理想关节约束，这可能导致在模拟环境中出现与物理世界不一致的现象。

当手动求解这样一个系统时，该方法仅将运动描述为广义坐标的函数。这仍然得到 $6n_j$ 个方程（对于平面系统来说有 $3n_j$ 个方程），但已经保证理想的关节约束。 $6n_j$ 个方程线性依赖于 n_j 个广义坐标和 $5n_j$ 个关节约束力。对方程组进行代数操作允许我们抵消所有的约束力。

【实例 3.3.1】牛顿欧拉车摆系统

待

补充

3.4 拉格朗日法

3.4.1 介绍

推导系统运动方程的另一种常见方法是使用所谓的拉格朗日方法。它起源于物理学的子领域—分析力学 [1][4]，并且与达朗贝尔和哈密顿原理密切相关，因为它是用来描述物理系统运动的分析方法之一。该方法围绕三个基本概念：

- 1 广义坐标 \mathbf{q} 和广义速度 $\dot{\mathbf{q}}$ 的定义，它可以含有也可以不涉及系统所受约束信息。
- 2 被称为拉格朗日函数的标量函数 \mathcal{L} 。对于机械系统来说，就是系统在每一瞬间总动能 \mathcal{T} 和总势能 \mathcal{U} 之间的差：

$$\mathcal{L} = \mathcal{T} - \mathcal{U} \quad (3.26)$$

- 3 所谓的欧拉-拉格朗日方程，也称为第二类的欧拉-拉格朗日方程，适用于拉格朗日函数 \mathcal{L} 以及外部广义力的总和 τ ：

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right) - \left(\frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right) = \tau \quad (3.27)$$

在最普遍的情况下，拉格朗日是广义坐标 \mathbf{q} 和其速度 $\dot{\mathbf{q}}$ 的函数，并且它也可能对时间 t 有明确的依赖，因此我们将上述标量能量函数重新定义为 $\mathcal{T} = \mathcal{T}(t, \mathbf{q}, \dot{\mathbf{q}})$ 和 $\mathcal{U} = \mathcal{U}(t, \mathbf{q})$ ，因此 $\mathcal{L} = \mathcal{L}(t, \mathbf{q}, \dot{\mathbf{q}})$ 。

最后，该公式最显著的特性之一是能够从最终的运动方程中消除系统内部的所有相互作用力，此处与牛顿-欧拉公式刚好相反，因为牛顿-欧拉公式明确考虑了系统内力。为了应用该方法导出复杂多体系统的运动方程，还必须考虑其他方面。这些将在本节末尾的简明概述中介绍，并在后文中立即进行解释。

3.4.2 动能

由 n_b 个刚体组成的系统的动能被定义为：

$$\mathcal{T} = \sum_{i=1}^{n_b} \left(\frac{1}{2} m_i \mathbf{A} \dot{\mathbf{r}}_{S_i}^T \mathbf{A} \dot{\mathbf{r}}_{S_i} + \frac{1}{2} {}^B \boldsymbol{\Omega}_{S_i}^T \cdot {}^B \boldsymbol{\Theta}_{S_i} \cdot {}^B \boldsymbol{\Omega}_{S_i} \right) \quad (3.28)$$

对于该系统中每个刚体 \mathcal{B}_i 来说，线速度部分可以在坐标系 \mathcal{A} 中表示计算，但旋转动能的计算在惯量矩阵 Θ_{S_i} 具有对角线形式的坐标系 \mathcal{B} 中计算更加方便，其中坐标系 \mathcal{B} 相对于坐标系 \mathcal{A} 旋转，并且坐标系 \mathcal{B} 的基向量是相对于质量原则分布。只要线速度 ${}^{\mathcal{A}}\dot{\mathbf{r}}_{S_i}$ 和角速度 ${}^{\mathcal{B}}\Omega_{S_i}$ 表示的是刚体的绝对速度，绝对速度在这里可以理解成相对于惯性坐标系的速度，公式 3.28 将得到正确的结果。

我们现在需要将动能表示为广义量的函数。为了实现这一点，我们使用公式 2.163 和公式 2.164 所描述的雅可比矩阵，但我们对每个刚体 \mathcal{B}_i 都进行了计算而不仅仅是末端执行器。所以我们能得到以下运动学关系：

$$\dot{\mathbf{r}}_{S_i} = \mathbf{J}_{S_i} \dot{\mathbf{q}} \quad (3.29)$$

$$\Omega_{S_i} = \mathbf{J}_{R_i} \dot{\mathbf{q}} \quad (3.30)$$

将这些关系代入公式 3.28 中动能的定义公式，得到以广义坐标表示的动能公式：

$$\mathcal{T}(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \underbrace{\left(\sum_{i=1}^{n_b} (\mathbf{J}_{S_i}^T m \mathbf{J}_{S_i} + \mathbf{J}_{R_i}^T \Theta_{S_i} \mathbf{J}_{R_i}) \right)}_{\mathbf{M}(\mathbf{q})} \dot{\mathbf{q}} \quad (3.31)$$

下划线表示的 $\mathbf{M}(\mathbf{q})$ 被定义为广义质量矩阵或广义惯量矩阵，正如我们将在后文中看到的，其对应于在最终运动方程中的惯量和非线性离心力与科里奥利力项。

3.4.3 势能

在典型的力学问题中，系统中的势能有两个基本来源；质量元器件的重力势能和弹性元器件从静止到变形时存储的能量。

在第一种情况下，每个刚体 \mathcal{B}_i 由于地球的重力势场的影响，保持势能。虽然重力势能在大尺度上不是线性的，但在大多数情况下，我们可以通过单位向量 ${}^I\mathbf{e}_g$ 定义的作用于质心的均匀单向势场来近似其对每个刚体的影响。如果我们知道刚体质心的位置 \mathbf{r}_{S_i} ，我们可以通过计算得到每个刚体的势能如下：

$$\mathbf{F}_{g_i} = m_i g {}^I\mathbf{e}_g \quad (3.32)$$

$$\mathcal{U}_g = - \sum_{i=1}^{n_b} \mathbf{r}_{S_i}^T \mathbf{F}_{g_i} \quad (3.33)$$

请注意，零能级可以任意选择。势能除了重力势能之外，许多应用还涉及包括弹簧或其他柔顺部件在内的弹性元件所对应的弹性势能。如果元器件 E_j 可以合理地近似为变形-力或变形-扭矩的线性关系，那么对应的势能可以通过以下关系

来描述：

$$\mathcal{U}_{E_j} = \frac{1}{2}k_j (d(\mathbf{q}) - d_0)^2 \quad (3.34)$$

其中 $d(\mathbf{q})$ 将弹性元件的瞬时位形 (例如弹簧的长度) 表示为广义坐标的函数。 d_0 定义为静止时的位形, 此时元件不受力, 例如线性弹簧的静止长度。我们将当前长度与静止长度之差 $(d(\mathbf{q}) - d_0)$ 称之为弹性元件的变形量。注意: 弹簧力可以等效地视为外力, 其大小和方向可能取决于广义坐标 \mathbf{q} 。这种形式类似于驱动器的扭矩。

3.4.4 外力

在系统上做功的所有外力都由广义力向量 τ 表示, 有关推导请参阅第 3.5.3 小节。

3.4.5 附加约束

我们必须考虑的最后一种系统是受到附加约束的系统。在大多数情况下, 当我们定义广义坐标时, 我们的目标是使用一组最小坐标来描述系统, 以便表达我们可以控制或至少测量的自由度。

考虑第 2.10.4 节中提出的案例, 接触约束会施加在基于浮动基座的系统上。这些约束可能不会一直处于活动状态, 例如, 步行机器人或在桌子上弹跳的球所受到的接触约束。通常, 当我们定义广义向量 \mathbf{q} 时, 这种接触约束不能被明确地考虑, 因为它可能并不总是时刻出现在系统中。克服这一点的最常见方法是使用拉格朗日乘子, 通常用向量 λ 表示。使用速度明确定义的约束被称为运动约束, 通常表示为广义速度的线性组合:

$$\sum_{k=1}^{n_q} a_{k,j}(\mathbf{q})\dot{q}_k + a_{0,j}(t), \quad j = 1, 2, \dots, n_{c,m} \quad (3.35)$$

对于 $n_{c,m}$ 个运动约束来说, 我们可以将所有 $a_{k,j}(\mathbf{q})$ 的系数堆叠起来, 从而得到运动约束下的雅可比矩阵 \mathbf{J}_m :

$$\mathbf{J}_m = \begin{bmatrix} a_{1,1} & \dots & a_{1,n_{c,m}} \\ \vdots & \dots & \vdots \\ a_{n_q,1} & \dots & a_{n_q,n_{c,m}} \end{bmatrix} \quad (3.36)$$

另一方面, 在某些情况下, 约束将被定义为位形的函数, 比如说标量函数 $f_j(\mathbf{q}) : \mathbb{R}^{n_q} \rightarrow \mathbb{R}$, 也就是位形约束。在这些情况下, 计算标量函数相对于广义坐标的梯度会得到第 2.10.4 节中描述的约束雅可比矩阵。

总的来说, 引入这些约束会得到描述约束的运动方程, 因为在 \mathbf{q} 的定义中没有说明附加约束, 而参数 $\boldsymbol{\lambda}$ 刚好能够引入附加约束。因此, 最终的运动方程是使用受约束的欧拉-拉格朗日 (CEL) 方程 (也称为第一类的欧拉-拉格朗日方程) 导出的:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right) - \left(\frac{\partial \mathcal{L}}{\partial \mathbf{q}} \right) + \mathbf{J}_m^T \boldsymbol{\lambda}_m + \left(\frac{\partial \mathbf{f}_c}{\partial \mathbf{q}} \right)^T \boldsymbol{\lambda}_c = \boldsymbol{\tau} \quad (3.37)$$

其中 $\boldsymbol{\lambda}_m \in \mathbb{R}^{n_{c,m}}$ 是运动约束的拉格朗日乘子向量, 而 $\boldsymbol{\lambda}_c \in \mathbb{R}^{n_{c,c}}$ 是用于位形约束的拉格朗日乘子向量。

总之, 我们应该清楚为什么拉格朗日方法也被称为基于能量的方法, 因为其运动方程是从标量能量函数导出的, 因此, 我们没有明确使用任何笛卡尔向量。所有运动学信息都被编码到广义坐标的定义中, 然后将其用于定义标量拉格朗日函数, 接着代入公式 3.27 以产生运动方程。这与明确处理笛卡尔向量的牛顿-欧拉方法形成鲜明对比。

3.5 投影牛顿-欧拉法

3.5.1 介绍

在本小节我们将介绍推导多体系统运动方程的最后一个方法, 即利用投影牛顿-欧拉 (PNE) 公式的方法。从本质上讲, 该方法是将笛卡尔坐标中动力学平衡的经典牛顿-欧拉方程与使用广义坐标表示的满足约束的拉格朗日公式相结合。事实上, 正如我们将在后续内容中将看到的, 我们可以从经典牛顿-欧拉方程公式和拉格朗日公式中导出投影牛顿-欧拉 (PNE) 方程。

让我们简单地预告一下使用投影牛顿-欧拉 (PNE) 公式的推导出来的运动方程将会是什么样子:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{J}_c^T \mathbf{F}_c \quad (3.38)$$

3.5.2 导出广义运动方程

如第 3.2.4 小节所述, 广义坐标的使用让我们可以描述满足系统约束的运动和虚位移。因此, 我们可以将应用于每个物体 \mathcal{B}_i 的线动量和角动量原理重写成:

$$\begin{aligned} \begin{pmatrix} \dot{\mathbf{p}}_{S_i} \\ \dot{\mathbf{N}}_{S_i} \end{pmatrix} &= \begin{pmatrix} m\mathbf{a}_{S_i} \\ \boldsymbol{\Theta}_{S_i} \boldsymbol{\Psi}_{S_i} + \boldsymbol{\Omega}_{S_i} \times \boldsymbol{\Theta}_{S_i} \boldsymbol{\Omega}_{S_i} \end{pmatrix} \\ &= \begin{pmatrix} m\mathbf{J}_{S_i} \\ \boldsymbol{\Theta}_{S_i} \mathbf{J}_{R_i} \end{pmatrix} \ddot{\mathbf{q}} + \begin{pmatrix} m_{S_i} \dot{\mathbf{J}}_{S_i} \dot{\mathbf{q}} \\ \boldsymbol{\Theta}_{S_i} \dot{\mathbf{J}}_{R_i} \dot{\mathbf{q}} + \mathbf{J}_{R_i} \dot{\mathbf{q}} \times \boldsymbol{\Theta}_{S_i} \mathbf{J}_{R_i} \dot{\mathbf{q}} \end{pmatrix} \end{aligned} \quad (3.39)$$

此外，使用广义坐标下的虚位移表达式，我们可以将虚功原理重写如下：

$$0 = \delta W = \sum_{i=1}^{n_b} \begin{pmatrix} \delta \mathbf{r}_{S_i} \\ \delta \Phi_{S_i} \end{pmatrix}^T \left(\begin{pmatrix} \dot{\mathbf{p}}_{S_i} \\ \dot{\mathbf{N}}_{S_i} \end{pmatrix} - \begin{pmatrix} \mathbf{F}_{\text{ext},i} \\ \mathbf{T}_{\text{ext},i} \end{pmatrix} \right) \quad \forall \begin{pmatrix} \delta \mathbf{r}_s \\ \delta \Phi_{S_i} \end{pmatrix}_{\text{consistent}}, \quad (3.40)$$

$$= \delta \mathbf{q}^T \sum_{i=1}^{n_b} \begin{pmatrix} \mathbf{J}_{S_i} \\ \mathbf{J}_{R_i} \end{pmatrix}^T \left(\begin{pmatrix} \dot{\mathbf{p}}_{S_i} \\ \dot{\mathbf{N}}_{S_i} \end{pmatrix} - \begin{pmatrix} \mathbf{F}_{\text{ext},i} \\ \mathbf{T}_{\text{ext},i} \end{pmatrix} \right) \quad \forall \delta \mathbf{q} \quad (3.41)$$

结合公式 3.39 和公式 3.41 直接得到：

$$0 = \sum_{i=1}^{n_b} \begin{pmatrix} \mathbf{J}_{S_i} \\ \mathbf{J}_{R_i} \end{pmatrix}^T \begin{pmatrix} m\mathbf{J}_{S_i} \\ \Theta_{S_i}\mathbf{J}_{R_i} \end{pmatrix} \ddot{\mathbf{q}} + \begin{pmatrix} \mathbf{J}_{S_i} \\ \mathbf{J}_{R_i} \end{pmatrix}^T \begin{pmatrix} m\dot{\mathbf{J}}_{S_i}\dot{\mathbf{q}} \\ \Theta_{S_i}\mathbf{J}_{R_i}\dot{\mathbf{q}} + \mathbf{J}_{R_i}\dot{\mathbf{q}} \times \Theta_{S_i}\mathbf{J}_{R_i}\dot{\mathbf{q}} \end{pmatrix} - \begin{pmatrix} \mathbf{J}_{P_i} \\ \mathbf{J}_{R_i} \end{pmatrix}^T \begin{pmatrix} \mathbf{F}_{\text{ext},i} \\ \mathbf{T}_{\text{ext},i} \end{pmatrix} \quad (3.42)$$

将得到的各项重新分组得到公式 3.38 的组成部分：

$$\mathbf{M} = \sum_{i=1}^{n_b} (\mathbf{J}_{S_i}^T m\mathbf{J}_{S_i} + \mathbf{J}_{R_i}^T \Theta_{S_i}\mathbf{J}_{R_i}) \quad (3.43)$$

$$\mathbf{b} = \sum_{i=1}^{n_b} \left(\mathbf{J}_{S_i}^T m\dot{\mathbf{J}}_{S_i}\dot{\mathbf{q}} + \mathbf{J}_{R_i}^T (\Theta_{S_i}\dot{\mathbf{J}}_{R_i}\dot{\mathbf{q}} + \Omega_{S_i} \times \Theta_{S_i}\Omega_{S_i}) \right)_i \quad (3.44)$$

$$\mathbf{g} = \sum_{i=1}^{n_b} -\mathbf{J}_{S_i}^T \mathbf{F}_{g,i} \quad (3.45)$$

我们必须再次强调选择合适的坐标系来表达所有笛卡尔向量的重要性。我们应用了与第 3.4.2 小节中计算动能时相同的推理过程，在那个小节中我们便已经指出角速度和线速度必须是绝对的，即速度是相对于惯性坐标系测量的。线速度和角速度在哪个坐标系表达并不重要，但是仔细选择坐标系可以简化最终得到的表达式：

$$\mathbf{M} = \sum_{i=1}^{n_b} (\mathbf{A}\mathbf{J}_{S_i}^T \cdot m \cdot \mathbf{A}\mathbf{J}_{S_i} + \mathbf{B}\mathbf{J}_{R_i}^T \cdot \mathbf{B}\Theta_{S_i} \cdot \mathbf{B}\mathbf{J}_{R_i}) \quad (3.46)$$

$$\mathbf{b} = \sum_{i=1}^{n_b} \left(\mathbf{A}\mathbf{J}_{S_i}^T \cdot m \cdot \mathbf{A}\dot{\mathbf{J}}_{S_i} \cdot \dot{\mathbf{q}} + \mathbf{B}\mathbf{J}_{R_i}^T \cdot (\mathbf{B}\Theta_{S_i} \cdot \mathbf{B}\dot{\mathbf{J}}_{R_i} \cdot \dot{\mathbf{q}} + \mathbf{B}\Omega_{S_i} \times \mathbf{B}\Theta_{S_i} \cdot \mathbf{B}\Omega_{S_i}) \right) \quad (3.47)$$

$$\mathbf{g} = \sum_{i=1}^{n_b} (-\mathbf{A}\mathbf{J}_{S_i}^T \cdot \mathbf{A}\mathbf{F}_{g,i}) \quad (3.48)$$

3.5.3 外力和驱动

正如我们在第 3.4.4 小节中看到的,为了解释广义坐标表示的运动方程中的外力,我们可以使用对应的雅可比矩阵将笛卡尔坐标系中的力和扭矩投影到广义坐标空间上。后者将力的作用投射到做功的广义坐标的子空间上。我们现在将考虑如何有效地考虑多体系统上的作用力和扭矩。

假设我们知道有 $n_{f,ext}$ 个外力 $\mathbf{F}_j, j \in \{1, \dots, n_{f,ext}\}$ 和 $n_{m,ext}$ 个外部扭矩 $\mathbf{T}_k, k \in \{1, \dots, n_{m,ext}\}$ 作用于系统 (作用于任何刚体)。由于外力产生的广义力 $\boldsymbol{\tau}_{F,ext}$ 可以通过以下方式计算: 假设笛卡尔力 \mathbf{F}_j 作用于点 P_j 且这一点的平移雅可比是 $\mathbf{J}_{P,j}$, 然后可以得到广义力的计算公式:

$$\boldsymbol{\tau}_{F,ext} = \sum_{j=1}^{n_{f,ext}} \mathbf{J}_{P,j}^T \mathbf{F}_j \quad (3.49)$$

广义力 $\boldsymbol{\tau}_{T,ext}$ 同样可以通过他们所作用的刚体的旋转雅可比 $\mathbf{J}_{B,k,rot}$ 来投影每个笛卡尔扭矩, 可以得到:

$$\boldsymbol{\tau}_{T,ext} = \sum_{k=1}^{n_{m,ext}} \mathbf{J}_{R,k}^T \mathbf{T}_{ext,k} \quad (3.50)$$

最后,因为它们都表示在广义坐标空间中,外力和扭矩的作用可以简单地相加:

$$\boldsymbol{\tau}_{ext} = \boldsymbol{\tau}_{F,ext} + \boldsymbol{\tau}_{T,ext} \quad (3.51)$$

对于驱动力或扭矩作用在两个连杆之间的这种情况,我们只需要考虑这两个 (连续的) 连杆之间运动学关系。因此,在刚体 \mathcal{B}_{k-1} 和 \mathcal{B}_k 之间作用的驱动器对两连杆施加的力 $\mathbf{F}_{a,k}$ 和或扭矩 $\mathbf{T}_{a,k}$ 大小相等方向相反,即是一对相互作用力。因此,需要计算的是将力和扭矩投影到广义坐标 \mathbf{q} 空间的雅可比矩阵。

为此,我们知道当多个雅可比矩阵在同一参考系中表达时,它们可以简单地相加或者相减,根据我们之前对运动学的分析计算得到的 $\mathbf{J}_{S_{k-1}}$ 、 \mathbf{J}_{S_k} 、 $\mathbf{J}_{R_{k-1}}$ 和 \mathbf{J}_{R_k} , 然后我们可以计算出驱动器对广义力的贡献如下:

$$\boldsymbol{\tau}_{a,k} = (\mathbf{J}_{S_k} - \mathbf{J}_{S_{k-1}})^T \mathbf{F}_{a,k} + (\mathbf{J}_{R_k} - \mathbf{J}_{R_{k-1}})^T \mathbf{T}_{a,k} \quad (3.52)$$

在最简单实际上也是最常见的情况下,驱动器沿广义坐标 \mathbf{q}_j 的方向作用,作用力点和反作用力点处的雅可比的差与力或扭矩的乘积与提供了广义力向量 $\boldsymbol{\tau}_j$ 中的单个项。因此,总外部广义力向量仅仅是上述部分的简单组合, n_A 个关节驱动器和 n_B 个刚体连杆的外部广义力为:

$$\boldsymbol{\tau} = \sum_{k=1}^{n_A} \boldsymbol{\tau}_{a,k} + \boldsymbol{\tau}_{ext} \quad (3.53)$$

评论:

- 1 首先求得刚体的质心处的平移雅可比，然后应用公式 2.154，可以获得刚体上任意点的平移雅可比。
- 2 或者，可以将所有作用线不通过物质心的力转换为通过质心的等效的力/力矩对，从而直接使用质心处的平移和旋转雅可比。
- 3 驱动器提供的力和扭矩除了作用和反作用力都在系统内部之外，与其他外力和外扭矩没有什么不同。

3.6 方法之间的总结和关系

- 1 所有需要广义坐标的定义
- 2 他们都需要质心的雅可比矩阵
- 3 评估一个可以求得另一个方程

为了检查推导的正确性，我们可以观测总能量的时间变化，也就是哈密顿量 $\mathcal{H} = \mathcal{T} + \mathcal{U}$ 。

3.7 浮动基座系统的动力学

浮动基座系统的动力学方程为：

$$\mathbf{M}(\mathbf{q})\dot{\mathbf{u}} + \mathbf{b}(\mathbf{q}, \mathbf{u}) + \mathbf{g}(\mathbf{q}) = \mathbf{S}^T \boldsymbol{\tau} + \mathbf{J}_{ext}^T \mathbf{F}_{ext} \quad (3.54)$$

由以下物理量组成：

$\mathbf{M}(\mathbf{q})$	$\in \mathbb{R}^{n_q \times n_q}$	正交惯量矩阵
\mathbf{q}	$\in \mathbb{R}^{n_q}$	广义坐标
\mathbf{u}	$\in \mathbb{R}^{n_q}$	广义速度
$\dot{\mathbf{u}}$	$\in \mathbb{R}^{n_q}$	广义加速度
$\mathbf{b}(\mathbf{q}, \mathbf{u})$	$\in \mathbb{R}^{n_q}$	科式力与离心力项
$\mathbf{g}(\mathbf{q})$	$\in \mathbb{R}^{n_q}$	重力项
\mathbf{S}	$\in \mathbb{R}^{n_\tau \times n_q}$	驱动关节选择矩阵
$\boldsymbol{\tau}$	$\in \mathbb{R}^{n_\tau}$	施加在广义坐标方向上的广义扭矩
\mathbf{F}_{ex}	$\in \mathbb{R}^{n_c}$	施加的广义力
\mathbf{J}_{ext}	$\in \mathbb{R}^{n_c \times n_q}$	外部力作用点处的几何雅可比

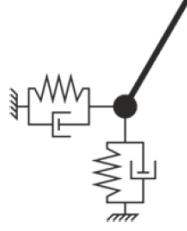


图 3.4: 在软接触模型中，接触力被建模为弹簧阻尼器力

正如我们在第 2.10 节中看到的，浮动基座系统的广义坐标由驱动关节坐标 \mathbf{q}_j 和欠驱动基坐标 \mathbf{q}_b 及对应的速度 $\mathbf{u}_j = \dot{\mathbf{q}}_j \in \mathbb{R}^{n_j}$ 和 $\mathbf{u}_b = \dot{\mathbf{q}}_b \in \mathbb{R}^6$ 组成。请再次注意 \mathbf{u}_b 不等于位置 and 方向参数化相对于时间的导数 $\dot{\mathbf{q}}$ ，因为不存在“角”位置这种概念，而存在不同方法来参数化方向。根据选择矩阵 \mathbf{S} 选择驱动关节：

$$\mathbf{u}_j = \mathbf{S}\mathbf{u} = \mathbf{S} \begin{pmatrix} \mathbf{u}_b \\ \mathbf{u}_j \end{pmatrix} = \begin{bmatrix} \mathbf{0}_{6 \times 6} & \mathbb{I}_{6 \times n_j} \end{bmatrix} \begin{pmatrix} \mathbf{u}_b \\ \mathbf{u}_j \end{pmatrix} \quad (3.55)$$

为了控制未驱动的基坐标 \mathbf{q}_b ，外力 \mathbf{F}_{ext} 是必须的。根据机器人的类型，外力可能的来源大不相同。例如，外力可以源于机器人与环境的接触产生的相互作用力（例如有腿的机器人）或在空气动力学中的升力（例如飞行的机器人）。在许多教科书中，在使用接触力时（例如腿足式机器人），人们常使用替代公式来表示系统动力学模型：

$$\mathbf{M}(\mathbf{q})\dot{\mathbf{u}} + \mathbf{b}(\mathbf{q}, \mathbf{u}) + \mathbf{g}(\mathbf{q}) + \mathbf{J}_c^T \mathbf{F}_c = \mathbf{S}^T \boldsymbol{\tau} \quad (3.56)$$

其中 \mathbf{F}_c 代表机器人对其环境施加的力。

3.7.1 接触力

存在两种完全不同的方法来模拟接触力。软接触方法使用力元件（比如说弹簧减震器）来对接触进行模拟，此时力是只关于接触点位置和速度的函数。硬接触方法将接触视为运动约束。

软接触模型

对于软接触模型，我们通常将首次与环境接触时的点定为 \mathbf{r}_{c0} 。当使用线性弹簧阻尼器对环境进行建模时，由机器人施加在环境的接触力为：

$$\mathbf{F}_c = k_p (\mathbf{r}_c - \mathbf{r}_{c0}) + k_d \dot{\mathbf{r}}_c \quad (3.57)$$

第一眼看来，对环境这样建模在逻辑和物理上似乎是正确的，但事实证明，找到物理上正确的弹簧和阻尼参数来模拟系统动力学几乎是不可能的。最大的问题就是多体系统与环境接触产生的组合微分方程刚度非常大（慢多体动力学和非常快的接触动力学）。对这些问题求解要么求解速度慢，要么求解结果精度低。为了克服这个问题，接触刚度和阻尼参数通常被调整为与实际物理参数无关的数值模拟参数。

来自约束的接触力

如果我们不把接触力建模成力元件产生的接触力，我们可以把接触作为运动学约束来处理。如果一个位于 \mathbf{r}_c 的点 C 与环境发生接触且不再允许移动：

$$\mathbf{r}_c = \text{const} \quad (3.58)$$

$$\dot{\mathbf{r}}_c = \mathbf{J}_c \mathbf{u} = 0 \quad (3.59)$$

$$\ddot{\mathbf{r}}_c = \mathbf{J}_c \dot{\mathbf{u}} + \dot{\mathbf{J}}_c \mathbf{u} = 0 \quad (3.60)$$

在点接触和扩展接触的情况下，可以对旋转约束应用相同的公式。从约束公式 3.60 和运动方程 3.56 中，我们可以确定接触力：

$$\mathbf{F}_c = (\mathbf{J}_c \mathbf{M}^{-1} \mathbf{J}_c^T)^{-1} \left(\mathbf{J}_c \mathbf{M}^{-1} (\mathbf{S}^T \boldsymbol{\tau} - \mathbf{b} - \mathbf{g}) + \dot{\mathbf{J}}_c \mathbf{u} \right) \quad (3.61)$$

这个公式能给我们带来很大的好处，因为它在没有任何接触力传感器的基础上，仅基于多体系统动力学，提供了地面反作用力的直接估计。

3.7.2 约束一致动力学

我们可以将动力学一致的支撑零空间矩阵定义为：

$$\mathbf{N}_c = \mathbb{I} - \mathbf{M}^{-1} \mathbf{J}_c^T (\mathbf{J}_c \mathbf{M}^{-1} \mathbf{J}_c^T)^{-1} \mathbf{J}_c \quad (3.62)$$

\mathbf{N}_c 定义了一个在支撑连杆上没有加速度或力耦合效应的广义运动空间。将接触力方程 3.61 的解代入运动方程 3.56 会得到：

$$\mathbf{M} \dot{\mathbf{u}} + \mathbf{N}_c^T (\mathbf{b} + \mathbf{g}) + \mathbf{J}_c^T (\mathbf{J}_c \mathbf{M}^{-1} \mathbf{J}_c^T)^{-1} \mathbf{J}_c \mathbf{u} = \mathbf{N}_c^T \mathbf{S}^T \boldsymbol{\tau} \quad (3.63)$$

进一步引入支撑约束公式 3.60 可得到 $\dot{\mathbf{J}}_c \mathbf{u} = -\mathbf{J}_c \dot{\mathbf{u}}$ ，约束一致运动方程可以紧凑地表述为：

$$\mathbf{N}_c^T (\mathbf{M} \dot{\mathbf{u}} + \mathbf{b} + \mathbf{g}) = \mathbf{N}_c^T \mathbf{S}^T \boldsymbol{\tau} \quad (3.64)$$

3.7.3 接触开关和碰撞

硬接触模型需要将系统动力学分析细分为两个阶段，分别对应于碰撞之前和之后。碰撞本身是一种复杂的物理现象，当两个或更多的刚体相互接触时就会发生。碰撞的特征是持续时间非常短、峰值力很大、能量快速消散和加速度大。为了模拟能量转移和耗散的过程，我们引入了各种系数，例如恢复系数和动量比系数。理想化这一过程，对应于将碰撞视为一个非常短暂的过程，也就需要考虑刚体接触时速度的瞬时变化。

动量转移

为了处理接触动量，我们在单个时间点 t_0 上的运动方程求积分：

$$\int_{\{t_0\}} (\mathbf{M}\dot{\mathbf{u}} + \mathbf{b} + \mathbf{g} + \mathbf{J}_c^T \mathbf{F}_c - \mathbf{S}^T \boldsymbol{\tau}) dt = \mathbf{M}(\mathbf{u}^+ - \mathbf{u}^-) + \mathbf{J}_c^T \mathcal{F}_c = \mathbf{0}, \quad (3.65)$$

其中 \mathcal{F}_c 为动量力， \mathbf{u}^- 和 \mathbf{u}^+ 分别对应碰撞前后的广义速度。假设碰撞是满足牛顿碰撞定律的完美非弹性碰撞，所有被认为是碰撞部分的接触点都会瞬间停止运动 ($\dot{\mathbf{r}}_c^+ = \mathbf{J}_c \mathbf{u}^+ = 0$)。将这种后碰撞约束与运动方程的积分相结合，我们可以求解动量力：

$$\mathcal{F}_c = (\mathbf{J}_c \mathbf{M}^{-1} \mathbf{J}_c^T)^{-1} \mathbf{J}_c \dot{\mathbf{q}}^- = \boldsymbol{\Lambda}_c \dot{\mathbf{r}}_c^- \quad (3.66)$$

通过定义**动量 = 质量 · 速度**的基本力学来更详细地分析这种形式，我们将在支撑点得到的惯量定为所谓的末端执行器的惯量：

$$\boldsymbol{\Lambda}_c = (\mathbf{J}_c \mathbf{M}^{-1} \mathbf{J}_c^T)^{-1} \quad (3.67)$$

将公式 3.67 代入公式 3.66 得到广义速度的瞬时变化：

$$\Delta \mathbf{u} = \mathbf{u}^+ - \mathbf{u}^- = -\mathbf{M}^{-1} \mathbf{J}_c^T (\mathbf{J}_c \mathbf{M}^{-1} \mathbf{J}_c^T)^{-1} \mathbf{J}_c \mathbf{u}^- \quad (3.68)$$

再次使用之前为动力学一致支撑零空间投影矩阵 \mathbf{N}_c 引入的命名法，后碰撞广义速度由以下公式得到：

$$\mathbf{u}^+ = \left(\mathbb{I} - \mathbf{M}^{-1} \mathbf{J}_c^T (\mathbf{J}_c \mathbf{M}^{-1} \mathbf{J}_c^T)^{-1} \mathbf{J}_c \right) \mathbf{u}^- = \mathbf{N}_c \mathbf{u}^- \quad (3.69)$$

通过满足后碰撞接触约束获得的结果直观上是明确的：使用支撑零空间投影矩阵 \mathbf{N}_c 可以将碰撞前的速度 \mathbf{u}^- 投影到支撑一致的流形上。

能量损失

碰撞的瞬时变化总是与动能损失息息相关。我们可以在广义坐标中对能量损失进行量化，也可以将能量损失描述为末端执行器惯量和支撑点速度变化的函数：

$$E_{\text{loss}} = \Delta E_{\text{kin}} = -\frac{1}{2} \Delta \mathbf{u}^T \mathbf{M} \Delta \mathbf{u} \quad (3.70)$$

$$= -\frac{1}{2} \Delta \dot{\mathbf{r}}_c^T \mathbf{\Lambda}_c \Delta \dot{\mathbf{r}}_c = -\frac{1}{2} \dot{\mathbf{r}}_c^{-T} \mathbf{\Lambda}_c \dot{\mathbf{r}}_c^- \quad (3.71)$$

3.8 关节空间动力学控制

目前的工业机器人几乎完全依赖于关节位置控制，位置控制建立在 PID 控制器上，以独立调节机器人每个关节的位置或速度为目的。这样的控制器通过对驱动器和整个机器人中的干扰进行补偿，从而在理想情况下实现对期望运动的完美跟踪。通过额外器件来测量关节扭矩（例如通过用负载元件测量或者通过电机电流和驱动器模型来估计它），便能实现基于模型的负载补偿。

3.8.1 关节阻抗调节

在扭矩可控的驱动器系统中，关节位置反馈增益 \mathbf{k}_p 和关节速度反馈增益 \mathbf{k}_d 对应于关节刚度和阻尼，期望的驱动器扭矩可以由以下公式计算得到：

$$\boldsymbol{\tau}^* = \mathbf{k}_p (\mathbf{q}^* - \mathbf{q}) + \mathbf{k}_d (\dot{\mathbf{q}}^* - \dot{\mathbf{q}}), \quad (3.72)$$

其中 \mathbf{q}^* 和 $\dot{\mathbf{q}}^*$ 分别表示期望的关节位置和速度。当将此控制律应用于机械臂时，我们得到稳态跟踪误差为：

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{k}_p (\mathbf{q}^* - \mathbf{q}) + \mathbf{k}_d (\dot{\mathbf{q}}^* - \dot{\mathbf{q}}) \quad (3.73)$$

重力补偿

为了补偿稳态误差和调节关节阻抗，一种常见的方法是选择期望的驱动器扭矩为：

$$\boldsymbol{\tau}^* = \mathbf{k}_p (\mathbf{q}^* - \mathbf{q}) + \mathbf{k}_d (\dot{\mathbf{q}}^* - \dot{\mathbf{q}}) + \hat{\mathbf{g}}(\mathbf{q}) \quad (3.74)$$

其中 $\hat{\mathbf{g}}(\mathbf{q})$ 表示估计的重力项。不幸的是，由于在每个关节处看到的惯量会随机器人位形的变化而变化，因此必须根据工作区中的大多数位形来选择 PD 增益。当系统剧烈变化时，该方法会降低控制器的性能。即使进行了重力补偿，稳态误差最终仍然存在，因为系统从未被完全准确地建模。

逆动力学控制

解决上述缺点的一个简单方法是使用逆动力学控制方法，实现动力学解耦和运动控制需要的关节转矩为：

$$\boldsymbol{\tau}^* = \hat{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}}^* + \hat{\mathbf{b}}(\mathbf{q}, \dot{\mathbf{q}}) + \hat{\mathbf{g}}(\mathbf{q}) \quad (3.75)$$

其中 $\hat{\mathbf{M}}(\mathbf{q})$ 、 $\hat{\mathbf{b}}(\mathbf{q}, \dot{\mathbf{q}})$ 以及 $\hat{\mathbf{g}}(\mathbf{q})$ 分别对应于对物理量 $\mathbf{M}(\mathbf{q})$ 、 $\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})$ 以及 $\mathbf{g}(\mathbf{q})$ 的估计。如果建模完美，带有控制公式 3.75 的系统 3.1 的闭环动力学为：

$$\mathbb{I}\ddot{\mathbf{q}} = \ddot{\mathbf{q}}^* \quad (3.76)$$

换句话说，这种方法允许我们直接改造每个关节的解耦动力学。与公式 3.72 中引入的阻抗定律类似，一种常见的选择期望加速度的方法为：

$$\ddot{\mathbf{q}}^* = \mathbf{k}_p(\mathbf{q}^* - \mathbf{q}) + \mathbf{k}_d(\dot{\mathbf{q}}^* - \dot{\mathbf{q}}) \quad (3.77)$$

其对应于具有单位质量的线性质量-弹簧-阻尼系统。一个巨大的好处是反馈 \mathbf{k}_p 和 \mathbf{k}_d 的调整和选择在直观上变得清晰，因为它们代表解耦的质点振荡器的物理参数。因此，系统的本征频率和无量纲阻尼值为：

$$\omega = \sqrt{k_p}, \quad (3.78)$$

$$D = \frac{k_d}{2\sqrt{k_p}} \quad (3.79)$$

$D = 1$ 时为临界阻尼， $D > 1$ 时为过临界阻尼和 $D < 1$ 时为欠临界阻尼。控制器的柔顺性可以通过改变 k_p 来调节。例如，假设标称点周围的振荡频率应为 3 Hz，那么理想的控制增益 k_p 是 350。此外，临界阻尼则要求 $k_d = 37$ 。这为控制器增益的调参提供了一个很好的起始值。

3.9 任务空间动力学控制

到目前为止，我们只考虑了基于模型的关节空间控制，然而在大多数情况下，我们想要在任务空间中，即在世界固定的坐标系中，移动机械臂到一个特定的点。末端执行器（或任何其他点和连杆）的线性和旋转加速度通过几何雅可比耦合到广义加速度：

$$\dot{\mathbf{w}}_e = \begin{pmatrix} \ddot{\mathbf{r}} \\ \ddot{\boldsymbol{\omega}} \end{pmatrix}_e = \mathbf{J}_e \ddot{\mathbf{q}} + \dot{\mathbf{J}}_e \dot{\mathbf{q}} \quad (3.80)$$

3.9.1 多任务分解

与第 2.9.2 节中概述的运动学多任务控制类似，我们可以在完成多个任务的同时执行逆动力学。通过期望的任务空间加速度和相应的雅可比给定一组运动目标，我们可以以相同的优先级对待所有目标：

$$\ddot{\mathbf{q}} = \begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_{n_t} \end{bmatrix}^+ \left(\begin{bmatrix} \dot{\mathbf{w}}_1 \\ \vdots \\ \dot{\mathbf{w}}_{n_t} \end{bmatrix} - \begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_{n_t} \end{bmatrix} \dot{\mathbf{q}} \right) \quad (3.81)$$

如果某些任务具有更高的优先级，我们可以使用公式 2.207 中概述的递归算法来确定解：

$$\ddot{\mathbf{q}} = \sum_{i=1}^{n_t} \mathbf{N}_i \ddot{\mathbf{q}}_i, \quad \text{with} \quad \ddot{\mathbf{q}}_i = (\mathbf{J}_i \mathbf{N}_i)^+ \left(\mathbf{w}_i^* - \mathbf{J}_i \dot{\mathbf{q}} - \mathbf{J} \sum_{k=1}^{i-1} \mathbf{N}_k \dot{\mathbf{q}}_k \right) \quad (3.82)$$

其中 \mathbf{N}_i 是堆叠雅可比 $\bar{\mathbf{J}}_i = \begin{bmatrix} \mathbf{J}_1^T & \dots & \mathbf{J}_i^T \end{bmatrix}^T$ 的零空间投影。

3.9.2 末端执行器动力学

分析和控制末端执行器描述的一个有趣的方法是考虑末端动力学。为此，我们可以求解方程 3.1 得到 $\ddot{\mathbf{q}}$ ，然后代入公式 3.80：

$$\dot{\mathbf{w}}_e = \mathbf{J}_e \mathbf{M}^{-1} (\boldsymbol{\tau} - \mathbf{b} - \mathbf{g}) + \dot{\mathbf{J}}_e \dot{\mathbf{q}} \quad (3.83)$$

末端执行器力前乘雅可比转置得到的结果可以替代关节扭矩：

$$\boldsymbol{\tau} = \mathbf{J}_e^T \mathbf{F}_e \quad (3.84)$$

得到末端执行器动力学方程：

$$\boldsymbol{\Lambda}_e \dot{\mathbf{w}}_e + \boldsymbol{\mu} + \mathbf{p} = \mathbf{F}_e \quad (3.85)$$

其中各项含义如下：

$$\boldsymbol{\Lambda}_e = (\mathbf{J}_e \mathbf{M}^{-1} \mathbf{J}_e^T)^{-1} \quad (3.86)$$

$$\boldsymbol{\mu} = \boldsymbol{\Lambda}_e \mathbf{J}_e \mathbf{M}^{-1} \mathbf{b} - \boldsymbol{\Lambda}_e \dot{\mathbf{J}}_e \dot{\mathbf{q}} \quad (3.87)$$

$$\mathbf{p} = \boldsymbol{\Lambda}_e \mathbf{J}_e \mathbf{M}^{-1} \mathbf{g} \quad (3.88)$$

分别表示任务空间中的末端执行器惯量、离心与科式力项和重力项。更多详细信息，感兴趣的读者请参阅 [5]。

3.9.3 末端执行器运动控制

类似于关节空间逆动力学控制，我们可以使用逆形式的任务空间运动方程来定义期望的关节扭矩。结合公式 3.84 和公式 3.85 得到：

$$\boldsymbol{\tau}^* = \hat{\mathbf{J}}^T \left(\hat{\boldsymbol{\Lambda}}_e \dot{\mathbf{w}}_e^* + \hat{\boldsymbol{\mu}} + \hat{\mathbf{p}} \right) \quad (3.89)$$

我们将这种形式与控制策略结合起来用于求解得到期望的加速度 $\dot{\mathbf{w}}^*$ ：

$$\dot{\mathbf{w}}_e^* = \mathbf{k}_p \begin{pmatrix} \mathbf{r}_e^* - \mathbf{r}_e \\ \Delta \phi_e \end{pmatrix} + \mathbf{k}_d (\mathbf{w}_e^* - \mathbf{w}_e) \quad (3.90)$$

请注意，我们使用 $\Delta \phi_e$ 表示末端执行器的旋转误差。请记住，对于旋转误差，我们不能简单地对两次旋转做减法，而是需要遵循第 2.9.3 节中介绍的方法。对于小误差，我们可以利用近似方法来处理：

$$\Delta \phi_e = \begin{bmatrix} \mathbb{I} & 0 \\ 0 & \mathbf{E}_R \end{bmatrix} \begin{pmatrix} \mathbf{r}^* - \mathbf{r} \\ \boldsymbol{\chi}_R^* - \boldsymbol{\chi}_R \end{pmatrix} \quad (3.91)$$

其中的矩阵 $\mathbf{E}_R(\boldsymbol{\chi})$ 会因为旋转参数化的方法不同而不同。

替代符号

在许多论文中，虽然 \mathbf{J}_e 不一定是可逆的，但是你会经常发现 $\boldsymbol{\Lambda} = \mathbf{J}_e^{-T} \mathbf{M} \mathbf{J}_e^{-1}$ ，使用这个表示的原因是因为这简化了方程的代数操作。实际上公式 3.90 可以直接简化为：

$$\boldsymbol{\tau}^* = \mathbf{J}^T (\boldsymbol{\Lambda}_e \dot{\mathbf{w}}_e^* + \boldsymbol{\mu} + \mathbf{p}) \quad (3.92)$$

$$= \mathbf{J}^T \boldsymbol{\Lambda}_e \dot{\mathbf{w}}_e^* + \mathbf{J}^T \mathbf{J}_e^{-T} \mathbf{M} \mathbf{J}_e^{-1} \mathbf{J}_e \mathbf{M}^{-1} \mathbf{b} - \mathbf{J}^T \boldsymbol{\Lambda}_e \dot{\mathbf{J}}_e \dot{\mathbf{q}} + \mathbf{J}^T \mathbf{J}_e^{-T} \mathbf{M} \mathbf{J}_e^{-1} \mathbf{J}_e \mathbf{M}^{-1} \mathbf{g} \quad (3.93)$$

$$= \mathbf{J}^T \boldsymbol{\Lambda}_e \dot{\mathbf{w}}_e^* + \underbrace{\mathbf{b} - \mathbf{J}^T \boldsymbol{\Lambda}_e \dot{\mathbf{J}}_e \dot{\mathbf{q}}}_{\tilde{\mathbf{b}}} + \mathbf{g} \quad (3.94)$$

$$= \mathbf{J}^T \boldsymbol{\Lambda}_e \dot{\mathbf{w}}_e^* + \tilde{\mathbf{b}} + \mathbf{g} \quad (3.95)$$

后面的操作空间公式同样可以写作这种形式。

3.9.4 操作空间控制

在许多情况下，机器人需要在某些方向上施加力，而需要在其他方向上移动。一个典型的例子是清洁窗户，机器人在正常方向上施加特定的压力并在其

他方向上控制运动。另一个例子是将销插入孔中：销应沿销的方向移动，并且不应围绕销的主轴旋转，而围绕另外两个轴的力和力矩应保持为零。为了实现这一点，我们引入了所谓的操作空间控制。根据 Khatib 的工作 [5]，我们可以根据运动和力的方向分别定义选择矩阵 \mathbf{S}_M 和 \mathbf{S}_F ，产生组合控制问题：

$$\boldsymbol{\tau}^* = \hat{\mathbf{J}}^T \left(\hat{\mathbf{A}} \mathbf{S}_M \dot{\mathbf{w}}_e + \mathbf{S}_F \mathbf{F}_c + \hat{\boldsymbol{\mu}} + \hat{\mathbf{p}} \right) \quad (3.96)$$

按照文献 [5]，我们可以定义位置和方向的配置矩阵：

$$\boldsymbol{\Sigma}_p = \begin{bmatrix} \sigma_{px} & 0 & 0 \\ 0 & \sigma_{py} & 0 \\ 0 & 0 & \sigma_{pz} \end{bmatrix} \quad \boldsymbol{\Sigma}_r = \begin{bmatrix} \sigma_{rx} & 0 & 0 \\ 0 & \sigma_{ry} & 0 \\ 0 & 0 & \sigma_{rz} \end{bmatrix} \quad (3.97)$$

其中 σ_i 是一个二进制数，1 值意味着机器人会沿着线性方向或围绕特定轴自由运动，否则为 0。如果接触力处的坐标系相对于惯性坐标系旋转，并可以由旋转变换矩阵 \mathbf{C} 描述，我们需要对选择矩阵进行变换，两个选择矩阵 \mathbf{S}_F 和 \mathbf{S}_M 于是被定义为：

$$\begin{aligned} \mathbf{S}_M &= \begin{bmatrix} \mathbf{C}^T \boldsymbol{\Sigma}_p \mathbf{C} & 0 \\ 0 & \mathbf{C}^T \boldsymbol{\Sigma}_r \mathbf{C} \end{bmatrix} \\ \mathbf{S}_F &= \begin{bmatrix} \mathbf{C}^T (\mathbb{I}_3 - \boldsymbol{\Sigma}_p) \mathbf{C} & 0 \\ 0 & \mathbf{C}^T (\mathbb{I}_3 - \boldsymbol{\Sigma}_r) \mathbf{C} \end{bmatrix} \end{aligned} \quad (3.98)$$

3.10 浮动基座系统逆动力学

当处理浮动基座系统时，我们需要在执行逆动力学的同时确保接触约束公式 3.60 得到满足。给定期望加速度 $\dot{\mathbf{u}}_{\text{consistent}}^*$ ，我们可以反转约束一致运动方程 3.64，从而得到：

$$\boldsymbol{\tau}^* = (\mathbf{N}_c^T \mathbf{S}^T)^+ \mathbf{N}_c^T (\mathbf{M} \mathbf{u} + \mathbf{b} + \mathbf{g}) \quad (3.99)$$

请注意，此处我们需要采用伪逆 $(\mathbf{N}_c^T \mathbf{S}^T)^+$ 。回想下运动学中多任务的解决方案：根据求伪逆的矩阵的结构，存在一个允许修改 $\boldsymbol{\tau}^*$ 的零空间：

$$\boldsymbol{\tau}^* = (\mathbf{N}_c^T \mathbf{S}^T)^+ \mathbf{N}_c^T (\mathbf{M} \dot{\mathbf{u}}^* + \mathbf{b} + \mathbf{g}) + \mathcal{N}(\mathbf{N}_c^T \mathbf{S}^T) \boldsymbol{\tau}_0^* \quad (3.100)$$

与此同时，支撑一致的运动方程 $\mathbf{N}_c^T \mathbf{S}^T \boldsymbol{\tau}^* = \mathbf{N}_c^T (\mathbf{M} \dot{\mathbf{u}} + \mathbf{b} + \mathbf{g})$ 仍然有效。换句话说，存在着不同的关节力矩分布能得到相同的运动 $\dot{\mathbf{u}}^*$ 。事实上，不同的扭矩分布会导致接触力分布的变化。在多接触情境下，我们可以设计不更改机器人的合力和力矩的接触内力，此时不会有任何额外的加速度产生。当将公式 3.99 作为伪逆时，解是满足约束一致运动方程的最小二乘最小转矩 $\boldsymbol{\tau}^*$ 。

3.10.1 二次问题

同时控制多个操作空间目标问题有很多种方法。这些任务包括控制所选位置处（比如末端执行器，质心等）的运动和期望的接触力与关节力矩。

将操作空间控制理解成线性目标的最小二乘优化序列问题是一个很全面的方法。为了应对带优先级任务空间逆动力学，在这一节我们将介绍了对优化变量为 \mathbf{x} 的 n_T 个线性方程组的层级最小二乘优化方法：

$$\mathbf{A}_i \mathbf{x} = \mathbf{b}_i \quad (3.101)$$

优先级表示为 $i \geq 1$ ，堆叠在矩阵 \mathbf{A}_i 和向量 \mathbf{b}_i 中，其中 $i = 1$ 是最高优先级。正如后面将介绍的，运动任务和关节力矩和接触力任务可以构建成为以关节加速度和关节力矩为优化变量的线性问题。在我们提出的层级框架中，我们的目标是在不影响任务的优先级的前提下在最小二乘意义上保证每项任务完成地尽可能的好：

$$\min_{\mathbf{x}} \|\mathbf{A}_i \mathbf{x} - \mathbf{b}_i\|_2 \quad (3.102)$$

我们有不同的方法来解决这个问题，例如使用第 3.10.2 小节讨论的迭代零空间投影方法或将问题当作可以使用标准的求解器来求解的系列约束二次规划 (QP) 问题。

3.10.2 迭代零空间投影

低优先级任务不得影响任何较高优先级任务这一要求可转换成把 \mathbf{x} 定义为 \mathbf{x}_i 前乘高优先级任务的零空间投影矩阵 \mathbf{N}_i 的和，即如下操作：

$$\mathbf{x} = \sum_{k=1}^{n_T} \mathbf{N}_k \mathbf{x}_k \quad (3.103)$$

该零空间投影 \mathbf{N}_i 被定义为 $\mathbf{N}_i = \mathcal{N}\left(\begin{bmatrix} \mathbf{A}_1^T & \dots & \mathbf{A}_{i-1}^T \end{bmatrix}^T\right)$ ，其中 $\mathbf{N}_1 = \mathbb{I}$ 并满足：

$$\mathbf{A}_i \mathbf{N}_j = \mathbf{0} \quad \forall i < j \quad (3.104)$$

正如我们在运动学部分所看到的，存在着计算零空间投影的不同方法。利用属性 3.104，带优先级地最小化问题 3.102 可以代入 3.103 从而可以分解成单个任务进行求解，从而求得 \mathbf{x}_i ：

$$\mathbf{A}_i \mathbf{x} - \mathbf{b}_i = \mathbf{A}_i \sum_{k=1}^{n_T} \mathbf{N}_k \mathbf{x}_k - \mathbf{b}_i \quad (3.105)$$

$$\mathbf{x}_i = (\mathbf{A}_i \mathbf{N}_i)^+ \left(\mathbf{b}_i - \mathbf{A}_i \sum_{k=1}^{i-1} \mathbf{N}_k \mathbf{x}_k \right) \quad (3.106)$$

计算序列的优化过程为：

算法 2 层级最小二乘优化

```

1 :  $n_T$  = 任务数量                                ▷初始最优解
2 :  $\mathbf{x} = \mathbf{0}$                                        ▷初始零空间投影矩阵
3 :  $\mathbf{N}_1 = \mathbb{I}$ 
4 : for  $i = 1 \rightarrow n_T$  do
5 :    $\mathbf{x}_i = (\mathbf{A}_i \mathbf{N}_i)^+ (\mathbf{b}_i - \mathbf{A}_i \mathbf{x})$ 
6 :    $\mathbf{x} = \mathbf{x} + \mathbf{N}_i \mathbf{x}_i$ 
7 :    $\mathbf{N}_{i+1} = \mathcal{N} \left( \begin{bmatrix} \mathbf{A}_1^T & \dots & \mathbf{A}_i^T \end{bmatrix}^T \right)$ 
8 : end for

```

3.10.3 序列约束优化

层级最小二乘优化的每一步都对应于线性约束的二次优化任务，其中较低优先级不得擅自更改高优先级任务：

$$\begin{aligned} \min_{\mathbf{x}} \quad & \|\mathbf{A}_i \mathbf{x} - \mathbf{b}_i\|_2 \\ \text{s.t.} \quad & \underbrace{\begin{bmatrix} \mathbf{A}_1 \\ \vdots \\ \mathbf{A}_{i-1} \end{bmatrix}}_{\bar{\mathbf{A}}_{i-1}} \mathbf{x} - \underbrace{\begin{pmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_{i-1} \end{pmatrix}}_{\mathbf{b}_{i-1}} = \mathbf{c}. \end{aligned} \quad (3.107)$$

注意：只要 $\hat{\mathbf{A}}_{i-1}$ 列满秩，成本 \mathbf{c} 是 $\mathbf{0}$ 。

系列约束二次优化问题可以使用标准 QP 解算器解决。

3.10.4 浮动基座系统任务空间控制 QP

具有复杂行为的机器人系统可以被描述为优化变量由广义加速度 $\dot{\mathbf{u}}$ 、接触力 \mathbf{F}_c 和关节力矩 $\boldsymbol{\tau}$ 堆叠而成的向量的多任务控制问题：

$$\mathbf{x} = \begin{pmatrix} \dot{\mathbf{u}} \\ \mathbf{F}_c \\ \boldsymbol{\tau} \end{pmatrix} \quad (3.108)$$

请注意：虽然我们的公式使用的是浮动基座系统广义速度符号 \mathbf{u} ，但是这种方法也可用于固定基座系统。使用此优化变量，运动方程 $\mathbf{M}\mathbf{u} + \mathbf{b} + \mathbf{g} + \mathbf{J}_c^T \mathbf{F}_c = \mathbf{S}^T \boldsymbol{\tau}$ 可以构建成最小二乘问题：

$$\mathbf{A} = \begin{bmatrix} \hat{\mathbf{M}} & \hat{\mathbf{J}}_c^T & -\mathbf{S}^T \end{bmatrix} \quad \mathbf{b} = -\hat{\mathbf{b}} - \hat{\mathbf{g}} \quad (3.109)$$

为了在感兴趣的点处实现期望的加速度 $\mathbf{J}\dot{\mathbf{u}} + \ddot{\mathbf{J}}\mathbf{u} = \dot{\mathbf{w}}^*$ ，我们可以简单地定义任务：

$$\mathbf{A} = \begin{bmatrix} \hat{\mathbf{J}}_i & 0 & 0 \end{bmatrix} \quad \mathbf{b} = \dot{\mathbf{w}}^* - \ddot{\mathbf{J}}_i \mathbf{u} \quad (3.110)$$

当然，我们也有许多其他感兴趣的目标。在下文中，我们举一些例子，但还有更多等待读者自己去实现。如果我们要在特定的点处以力或力矩 $\mathbf{F}_i = \mathbf{F}_i^*$ 运动到特定的位置，我们可以定义以下任务：

$$\mathbf{A} = \begin{bmatrix} 0 & \mathbf{I} & 0 \end{bmatrix} \quad \mathbf{b} = \mathbf{F}_i^* \quad (3.111)$$

此外，如果我们想最大限度地减少整体关节力矩，我们可以定义以下任务：

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & \mathbf{I} \end{bmatrix} \quad \mathbf{b} = 0 \quad (3.112)$$

【实例 3.10.1】二次规划求解关节空间逆动力学

给定期望加速度 $\ddot{\mathbf{q}}^*$ ，请为不接触环境的固定基座机械臂构建关节空间逆动力学控制方法。说明：这是一个纯学术的例子，因为使用本节中介绍的优化方法来解决这个问题是大材小用的。

任务 1 运动方程：

$$\mathbf{A} = \begin{bmatrix} \hat{\mathbf{M}} & 0 & -\mathbf{I} \end{bmatrix} \quad \mathbf{b} = -\hat{\mathbf{b}} - \hat{\mathbf{g}} \quad (3.113)$$

任务 2 实现期望的关节空间加速度：

$$\mathbf{A} = \begin{bmatrix} \mathbf{I} & 0 & 0 \end{bmatrix} \quad \mathbf{b} = \ddot{\mathbf{q}}^* \quad (3.114)$$

3.11 准静态 (虚拟模型) 控制

注意：此部分不属于讲课内容，但仍提供给感兴趣的读者。

对于执行相对缓慢的动作的机器人来说, 在系统动力学中起主要作用的是外部载荷力。在这种情况下, 我们可以考虑忽略 $\mathbf{M}\ddot{\mathbf{q}}$ 和 $\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})$ 项的准静态运动方程。一种广泛使用的准静态控制方法是 Pratt 等人在文献 [6] 中引入的虚拟模型控制 (VMC)。此方法能生成与在系统上指定位置 \mathbf{r}_{v_i} 作用的外部 (虚拟) 力 \mathbf{F}_{v_i} 等效的期望转矩。为了导出浮动基座系统的 VMC 的推广形式, 我们应用虚功原理 (见 3.2.5), 也就意味着多体系统变分做的功必须抵消所有虚位移的虚功。已知浮动基座系统上任意点 p_i 的虚位移可以写成:

$$\delta \mathbf{r}_{p_i} = \delta \mathbf{r}_b + \delta \boldsymbol{\varphi}_b \times \mathbf{r}_{bp_i} + \delta \mathbf{r}_{bp_i} = \begin{bmatrix} \mathbb{I} & -[\mathbf{r}_{bp_i}]_{\times} & \mathbf{J}_{bp_i} \end{bmatrix} \begin{pmatrix} \delta \mathbf{r}_b \\ \delta \boldsymbol{\varphi}_b \\ \delta \mathbf{q}_r \end{pmatrix} \quad (3.115)$$

其中 $\delta \mathbf{r}_b$ 和 $\delta \boldsymbol{\varphi}_b$ 是基节点 b 相对于惯性系的位置和旋转的变分。反对称矩阵 $[\mathbf{r}_{bp_i}]_{\times} = -[\mathbf{r}_{bp_i}]_x^T$ 对应于叉乘矩阵 $[\mathbf{r}_{bp_i}]_{\times} \boldsymbol{\varphi}_b = \mathbf{r}_{bp_i} \times \boldsymbol{\varphi}_b$, 其中 \mathbf{r}_{bp_i} 是从基点 b 指向 p_i 的相对位置向量。相对位置变分 $\delta \mathbf{r}_{bp_i} = \mathbf{J}_{bp_i} \delta \mathbf{q}_r$ 使用广义关节坐标 $\delta \mathbf{q}_r$ 和投影相对雅可比 $\mathbf{J}_{bp_i} = \frac{\partial \mathbf{r}_{bp_i}}{\partial \mathbf{q}_r}$ 表达。在这种参数化下, 任意变分中外部和内力所产生虚功为:

$$\begin{aligned} \delta W &= \delta \mathbf{q}_r^T \boldsymbol{\tau} + \sum_i \delta \mathbf{r}_{p_i}^T \mathbf{F}_{p_i} \\ &= \begin{bmatrix} \delta \mathbf{r}_b^T & \delta \boldsymbol{\varphi}_b^T & \delta \mathbf{q}_r^T \end{bmatrix} \left(\begin{pmatrix} 0 \\ 0 \\ \boldsymbol{\tau} \end{pmatrix} + \sum_i \begin{bmatrix} 1 \\ [\mathbf{r}_{bp_i}]_{\times} \\ \mathbf{J}_{bp_i}^T \end{bmatrix} \mathbf{F}_{p_i} \right) = \mathbf{0} \quad \forall \begin{pmatrix} \delta \mathbf{r}_b \\ \delta \boldsymbol{\varphi}_b \\ \delta \mathbf{q}_r \end{pmatrix}, \\ \rightarrow 0 &= \sum_i \mathbf{F}_{p_c}, \end{aligned} \quad (3.116)$$

$$\rightarrow 0 = \sum_i \mathbf{r}_{bp_i} \times \mathbf{F}_{p_t} \quad (3.117)$$

$$\rightarrow 0 = \boldsymbol{\tau} + \sum_i \mathbf{J}_{bp_i}^T \mathbf{F}_{p_u} \quad (3.118)$$

其中 $\sum_i \mathbf{F}_{p_i} = \sum_i \mathbf{F}_{g_i} - \sum_i \mathbf{F}_{v_i} - \sum_i \mathbf{F}_{c_i}$ 表示包括重力 (\mathbf{F}_{g_i})、虚拟控制力 ($-\mathbf{F}_{v_i}$) 和接触力 ($-\mathbf{F}_{c_i}$) 在内的所有外力。方程 3.116 和 3.117 对应于平衡的所有外部载荷的力和力矩, 并可以用于确定未知地面接触力 \mathbf{F}_{c_i} 。在大多数情况下, 这是根据以下公式通过伪逆得到:

$$\begin{pmatrix} \mathbf{F}_{c_1} \\ \vdots \\ \mathbf{F}_{c_{n_c}} \end{pmatrix} = \begin{bmatrix} \mathbb{I} & \cdots & \mathbf{I} \\ [\mathbf{r}_{c_1}]_{\times} & \cdots & [\mathbf{r}_{c_{m_e}}]_{\times} \end{bmatrix}^+ \begin{bmatrix} \sum \mathbf{F}_{g_i} - \sum \mathbf{F}_{v_i} \\ \sum \mathbf{r}_{g_i} \times \mathbf{F}_{g_i} - \sum \mathbf{r}_{v_i} \times \mathbf{F}_{v_i} \end{bmatrix} \quad (3.119)$$

考虑到所有这些外力，期望的转矩可以从公式 3.118 得到：

$$\boldsymbol{\tau} = - \sum_i \mathbf{J}_{bg_i}^T \mathbf{F}_{g_i} + \sum_i \mathbf{J}_{bv_i}^T \mathbf{F}_{v_i} + \sum_i \mathbf{J}_{bc_i}^T \mathbf{F}_{c_i} \quad (3.120)$$

为了简化表示，我们没有包含外部扭矩 $\boldsymbol{\tau}_i^a$ ，但是他们可以和相应的旋转雅可矩阵相乘得到 $\sum_i \mathbf{J}_{\mathbf{R}_i}^T \boldsymbol{\tau}_i^a$ ，然后直接添加到公式中。

参考文献

- [1] H Baruh. Analytical Dynamics (WCB/McGraw-Hill). 1999.
- [2] Michael Bloesch et al. “A primer on the differential calculus of 3d orientations”. In: arXiv preprint arXiv:1606.05285 (2016).
- [3] Samuel R Buss. “Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods”. In: IEEE Journal of Robotics and Automation 17.1-19 (2004), p. 16.
- [4] JL Safko H. Goldstein CP Poole. Classical Mechanics. Addison Wesley, 2002.
- [5] Oussama Khatib. “A unified approach for motion and force control of robot manipulators: The operational space formulation”. In: IEEE Journal on Robotics and Automation 3.1 (1987), pp. 43–53.
- [6] Jerry Pratt et al. “Virtual model control: An intuitive approach for bipedal locomotion”. In: The International Journal of Robotics Research 20.2 (2001), pp. 129–143.
- [7] Bruno Siciliano, Oussama Khatib, and Torsten Kröger. Springer handbook of robotics. Vol. 200. Springer, 2008.
- [8] Bruno Siciliano et al. “Modelling, planning and control”. In: Advanced Textbooks in Control and Signal Processing (2009).
- [9] Joan Solà Ortega. Quaternion kinematics for the error-state KF. 2016.