

第二章 机器人轨迹生成与运动控制

华东理工大学信息科学与工程学院

卿湘运

2024年1月

2 机器人轨迹生成与运动控制

2.1

机器人运动控制的概念和流程

2.2

机器人运动控制的实现

2.3

机器人运动控制问题和方法的分类

2.1 机器人运动控制的概念和流程

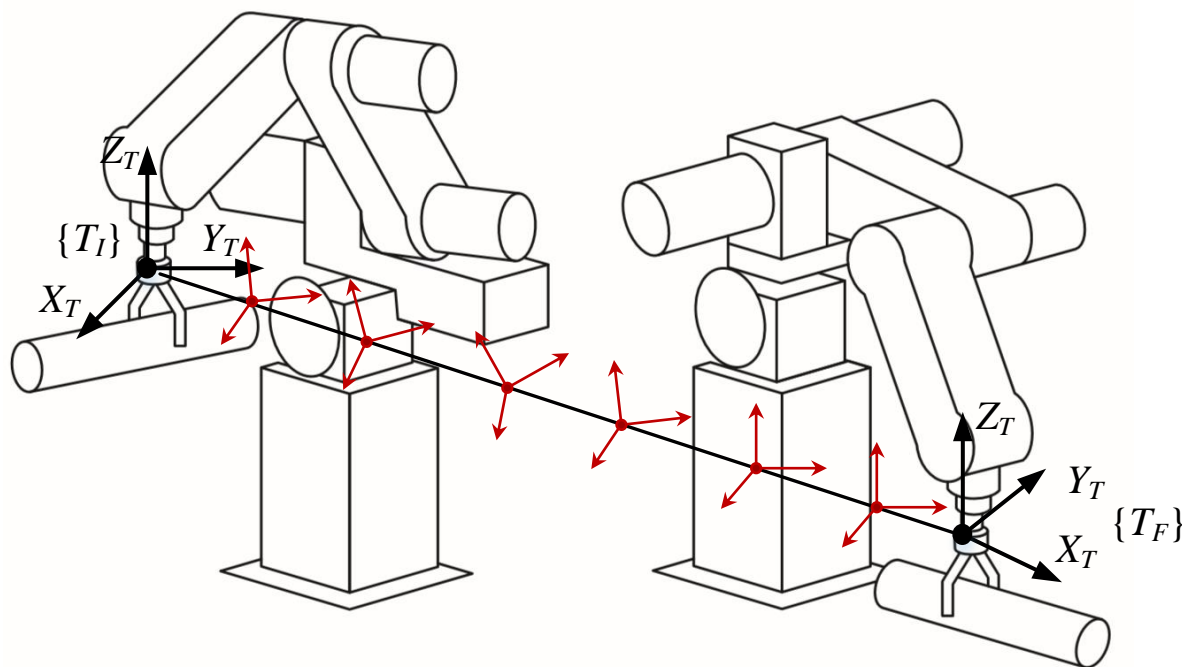
“运动控制”是机器人控制的基本问题，也是机器人控制系统需要解决的首要问题。机器人学里的**“运动”和“力”是对偶量**，它们的控制问题可以依托同一套软硬件系统来实现。

因此，实现位置、速度以及力控制的软硬件系统也被称为**运动控制系统** (motion control system)，或者伺服系统，机器人伺服系统的核心功能就是使被控量，例如关节或末端的位移或力，跟踪控制指令。

在工程实践中，不会要求机器人操作员直接给定伺服环指令，而仅需要操作员给机器人运动控制系统输入生成工艺信息。

2.1 机器人运动控制的概念和流程

怎样让机器人运动起来？



指定若干位姿



路径规划



生成位姿序列



生成关节序列



关节控制



机器人运动

2.1 机器人运动控制的概念和流程

2.1.1 几个重要概念:

- 路径 (Path)

几何概念，描述机器人末端在空间中走过的曲线，路径点的集合。

$$\boxed{\text{路径}} = \boxed{\text{点路径}} + \boxed{\text{姿态路径}}$$

- 路径点

路径上的一点，代表机器人末端位姿。

$$x = (p^T \quad \eta^T)^T = \overset{\text{位置矢量}}{\boxed{(x \quad y \quad z)}} \overset{\text{姿态矢量}}{\boxed{(\alpha \quad \beta \quad \gamma)}}^T$$

2.1 机器人运动控制的概念和流程

2.1.1 几个重要概念：

● 空间点路径

描述末端坐标系原点位置的变化过程。

$$\mathbf{p} = \mathbf{p}(s), \quad s \in [0, 1] \quad \left\{ \begin{array}{l} \bullet \mathbf{p} = [x(s), y(s), z(s)]^T : \text{位置} \\ \bullet s : \text{连续变化的标量, 描述动态过程} \\ \bullet s = 0 : \text{路径起点} \\ \bullet s = 1 : \text{路径终点} \end{array} \right.$$

➤ $\mathbf{p}(s)$ 建立了空间三维坐标的约束，反应了路径的形状。

2.1 机器人运动控制的概念和流程

2.1.1 几个重要概念:

● 姿态路径

描述末端坐标系姿态的变化过程。

$$\boldsymbol{\Omega} = \boldsymbol{\Omega}(u), \quad u \in [0, 1]$$

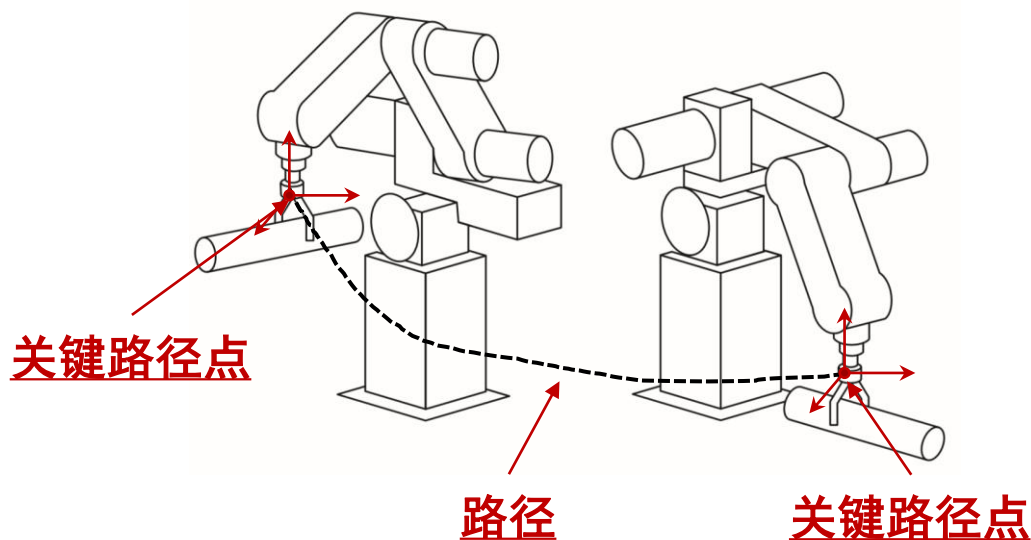
- $\boldsymbol{\Omega} = [\phi(u), \theta(u), \psi(u)]^T$: 姿态
- 三角度、等效轴角、单位四元数
- u : 与 s 类似的标量
- $u = s$: 位置与姿态同步变化

2.1 机器人运动控制的概念和流程

2.1.1 几个重要概念：

● 路径规划(Path Planning)

- 在机器人位形空间中生成一个**连续**的可行位形子空间
- 对工业机器人，就是生成连接已知关键路径点的空间路径



➤ 自主规划

- 六维空间的最优搜索问题
- 需要考虑避障、奇异等问题
- 本课程不研究

➤ 人工辅助规划

- 示教编程
- 离线编程

2.1 机器人运动控制的概念和流程

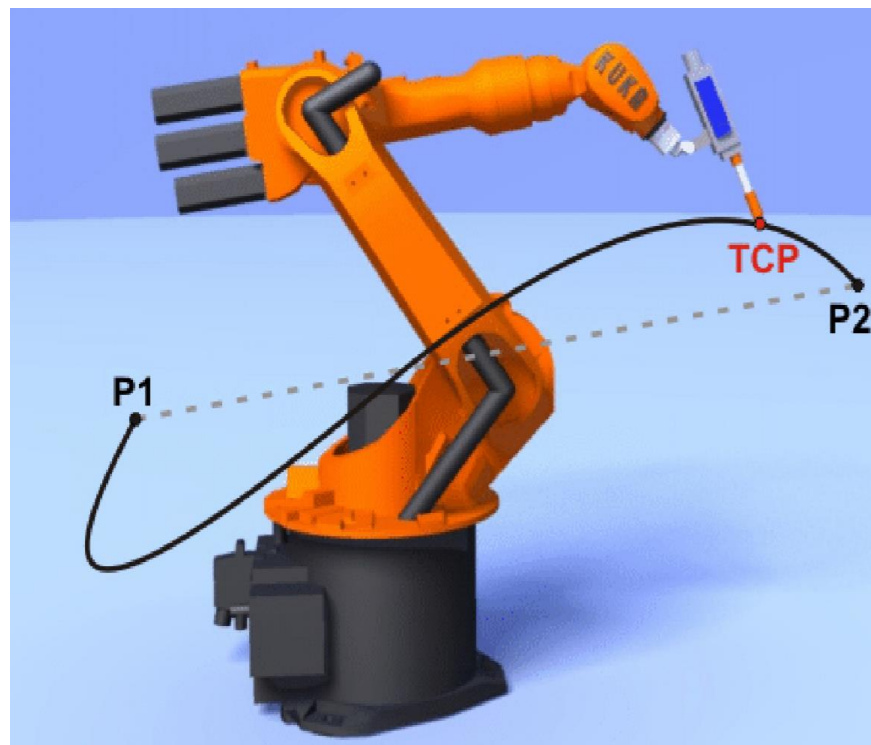
2.1.1 几个重要概念：

● 点到点（Point to Point—PTP）模式

➤ 指定末端起始位姿点、终止位姿点和少数中间关键位姿点，无需指定稀疏位姿点之间的连接路径

➤ 程序在关节空间生成关节轨迹

➤ 用途：上/下料、点焊、激光打孔等



2.1 机器人运动控制的概念和流程

2.1.1 几个重要概念：

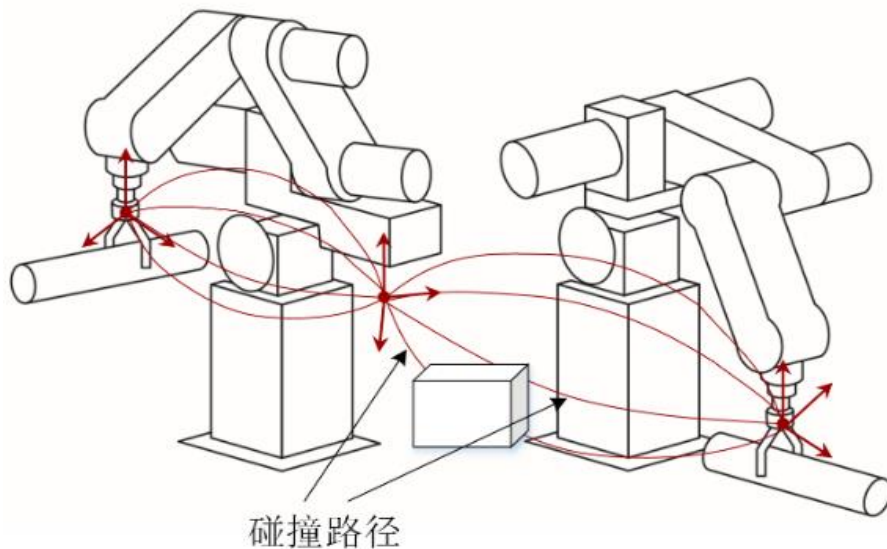
● 点到点（Point to Point—PTP）模式

➤ 优点

- 无需考虑奇异性问题
- 无需频繁计算运动学逆解，计算效率高

➤ 问题

- 只能准确到达几个稀疏的位姿点
- 关节变量与末端位姿往往存在非线性映射，无法预知实际运动路径

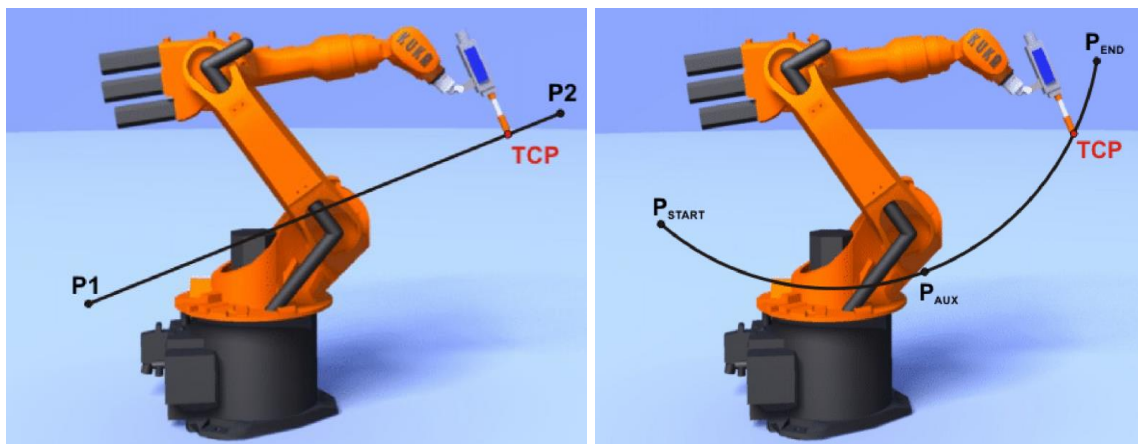


2.1 机器人运动控制的概念和流程

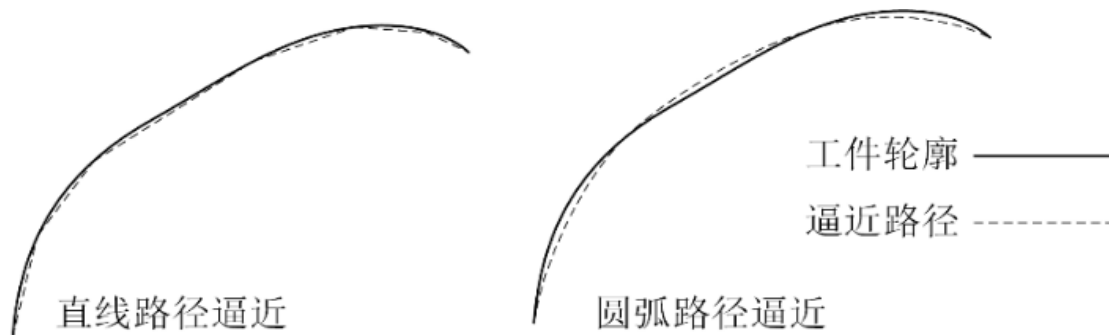
2.1.1 几个重要概念：

- 连续路径（Continuous Path—CP）模式

- 用指定的空间曲线连接关键位姿点，例如直线和圆弧



- 跟踪特定的空间曲线，通常用多个短直线或圆弧段进行逼近



2.1 机器人运动控制的概念和流程

2.1.1 几个重要概念：

● 连续路径（Continuous Path—CP）模式

➤ 优点

- 在真实物理空间中进行轨迹规划，机器人**运动行为可预测**
- 可跟踪**稠密位姿点**，适用于需要精确跟踪指定路径的场合
- 用途：打胶、喷涂、弧焊、切割、铣削等

➤ 问题

- 需要频繁进行逆解运算，**运算量大**，难实现高频率插补
- 为规避奇异点，要求用户了解机器人工作空间中的奇异点分布情况

2.1 机器人运动控制的概念和流程

2.1.1 几个重要概念：

● 轨迹

为机器人指定走完某一路径所需要的时间，就获得了轨迹。

$$\begin{aligned} & \mathbf{x}(t) = [\mathbf{p}^T(s(t)) \quad \boldsymbol{\eta}^T(u(t))]^T \\ \text{或} & \mathbf{x}(t) = [\mathbf{p}^T(t) \quad \boldsymbol{\eta}^T(t)]^T \end{aligned}$$

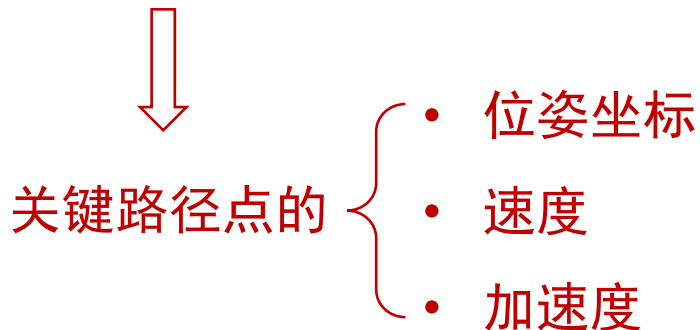
- 轨迹不但包含路径的几何信息，还包含速度和加速度等物理信息；
- 从数学上看，轨迹既是位姿点各分量间的几何约束函数，也是时间函数；
- 轨迹函数用统一的时变参数 $s(t)$ 或 $u(t)$ 表达，便于调整跟踪时的加减速规律。

2.1 机器人运动控制的概念和流程

2.1.1 几个重要概念：

● 轨迹生成

在满足**特定约束**的条件下，获得轨迹时间函数的过程



2.1 机器人运动控制的概念和流程

2.1.1 几个重要概念：

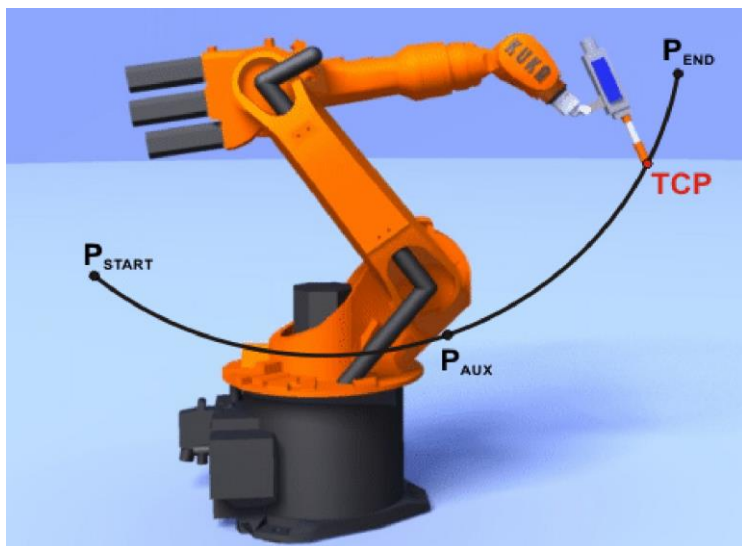
- 轨迹插补

对连续轨迹函数在时间上进行离散化，使计算机能实施离散控制

- 插补周期和插补点

➤ **插补周期**：离散时间间隔，在每个插补周期，计算机控制末端或关节走一个微小直线段轨迹；

➤ **插补点**：轨迹上的离散位姿值



微小直线段长度=

$$\boxed{\text{当前速度}} \times \boxed{\text{插补周期}}$$

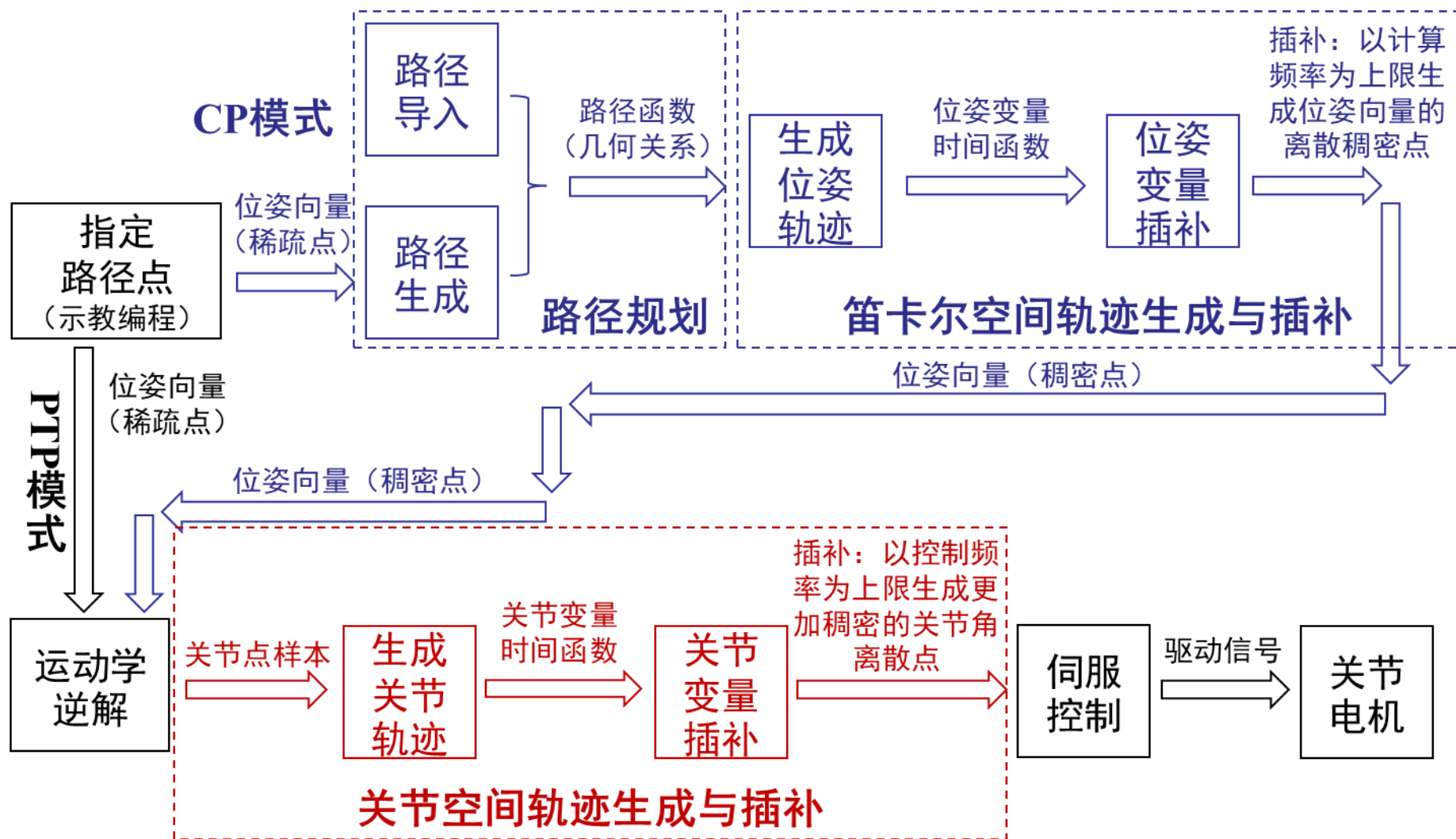
↙ ↘

$$\boxed{\text{插补精度}}$$

越短，精度越高！

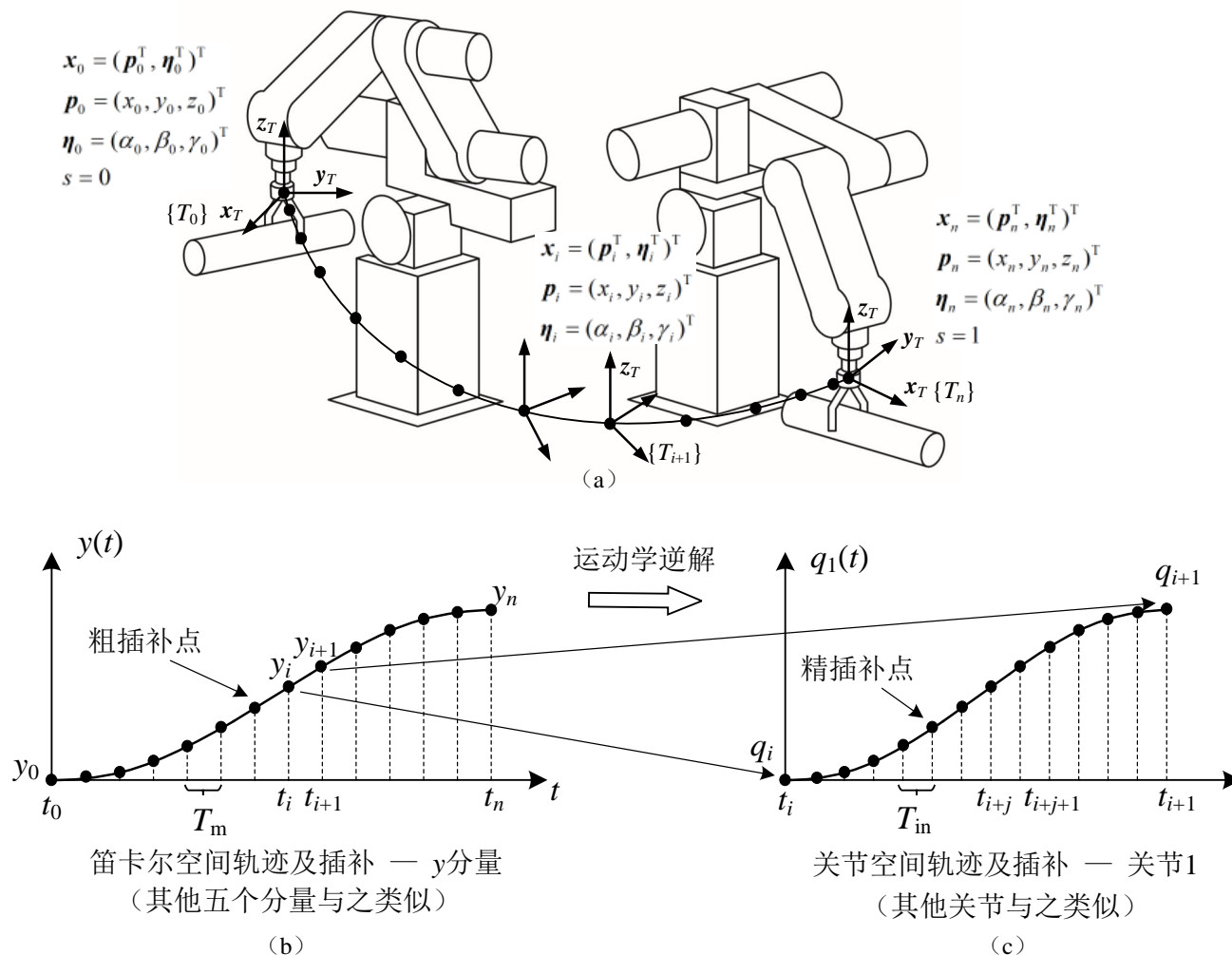
2.1 机器人运动控制的概念和流程

2.1.2 机器人运动控制的流程



2.1 机器人运动控制的概念和流程

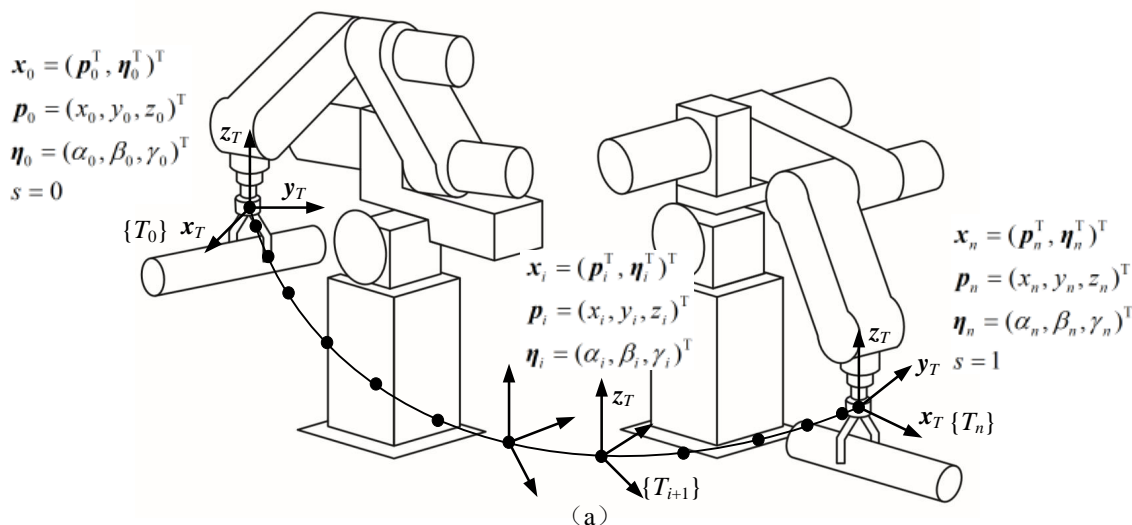
2.1.2 机器人运动控制的流程--轨迹生成和插补



2.1 机器人运动控制的概念和流程

2.1.2 机器人运动控制的流程--笛卡尔空间轨迹生成及插补

在笛卡尔空间指定路径跟踪时间，并据此生成轨迹的过程。



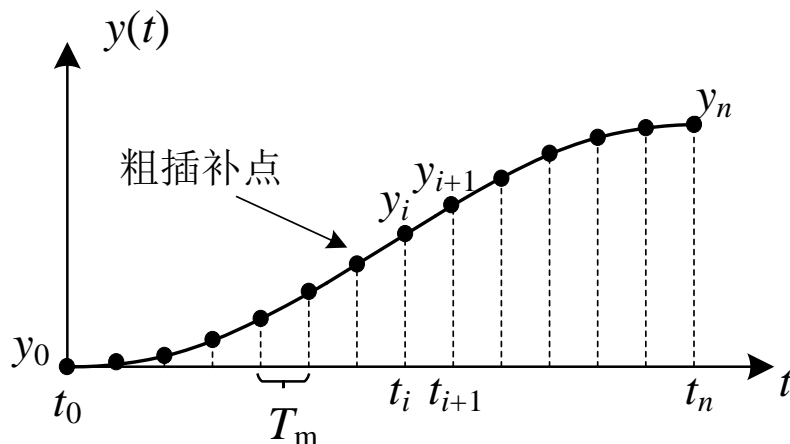
➤ 以圆弧路径为例：

- 指定路径点：跟踪圆弧路径需要采用CP模式，指定圆弧起点、终点和圆心坐标。
- 路径规划：利用圆弧路径生成算法，得到圆弧路径函数。如果采用离线编程，可以直接导入已知的圆弧路径函数
- 笛卡尔空间轨迹生成：把圆弧路径函数表示为参数方程 $x(t) = (p^T(s(t)) \quad \eta^T(u(t)))^T$ 的形式，然后指定参数 $s(t)$ 和 $u(t)$ 的时间函数

2.1 机器人运动控制的概念和流程

2.1.2 机器人运动控制的流程--笛卡尔空间轨迹生成及插补

□ 末端位姿向量中y分量的轨迹曲线和插补点坐标。



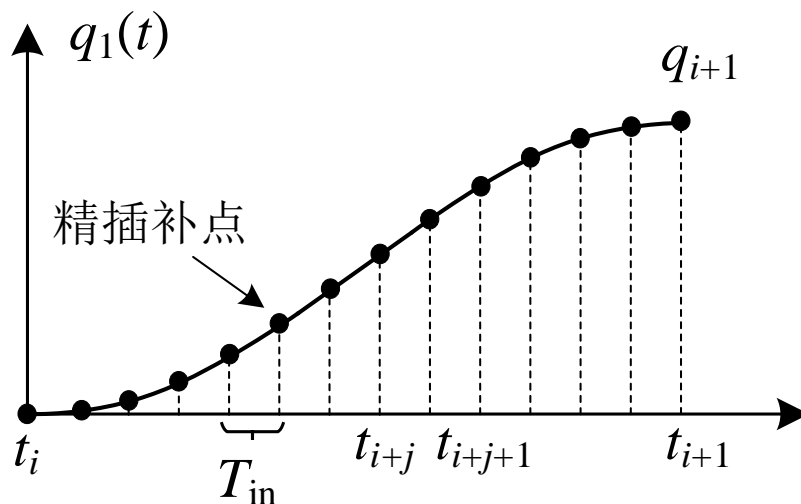
笛卡尔空间轨迹及插补 — y分量

- 需对各插补轨迹点求逆解，获得关节位置、速度和加速度
- 实时避障或变速要求逆解计算能实时在线进行
- 为缓解运算压力，插补周期相对较大（数十毫秒， Δt_c ），称为粗插补

2.1 机器人运动控制的概念和流程

2.1.2 机器人运动控制的流程—关节空间轨迹生成及插补

□ 对某关节，在已知关节位置之间生成关节变量时间函数的过程。

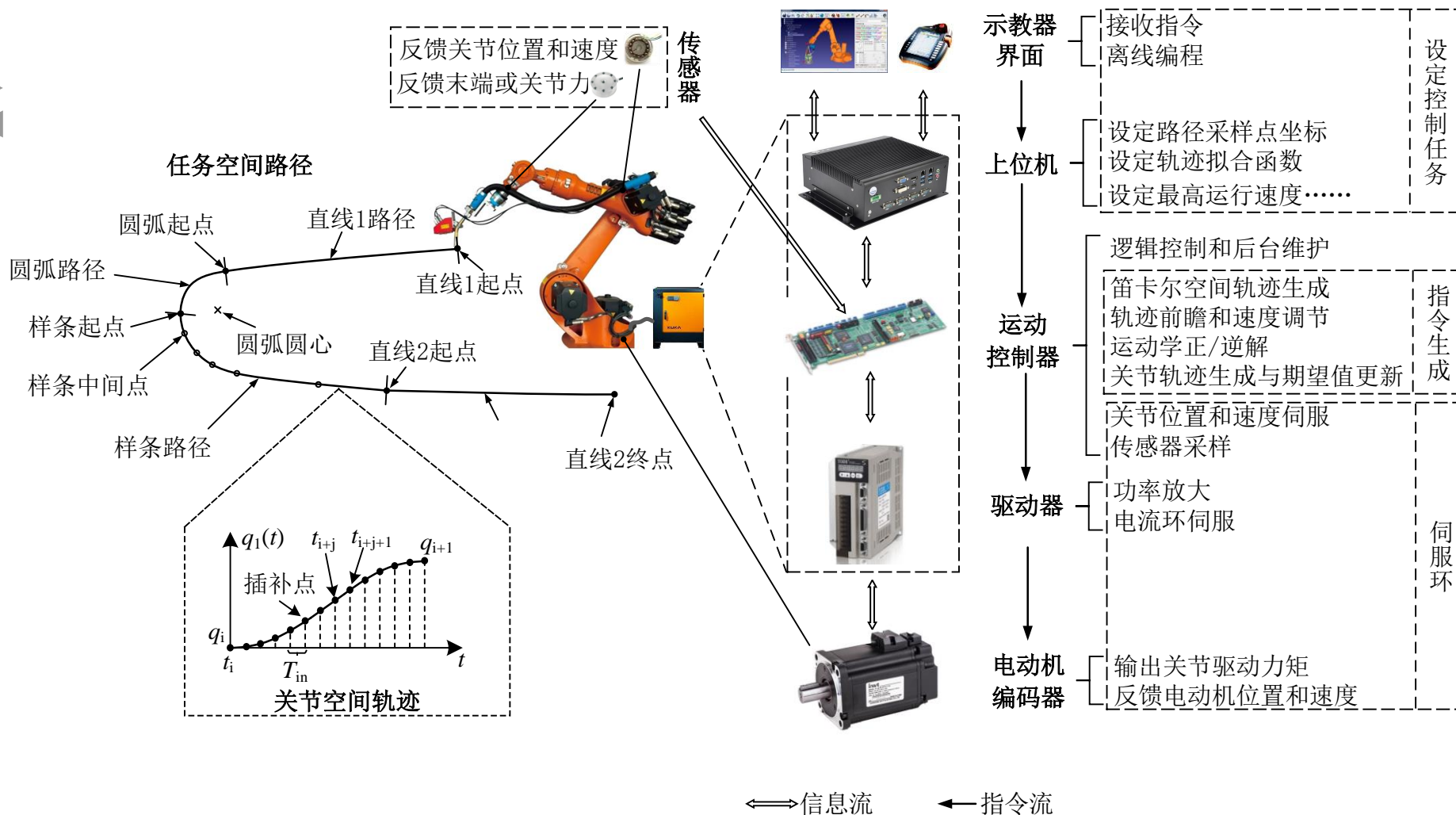


关节空间轨迹及插补 — 关节1

- 更小的关节轨迹插补周期（数十至数百微秒， Δt ），称为**精插补**
- 无论PTP还是CP模式，都需要进行关节轨迹生成和插补

2.2 机器人运动控制的实现

2.2.1 运动控制系统中的硬件——机器人运动控制系统组成和运行原理



2.2 机器人运动控制的实现

2.2.1 运动控制系统中的硬件——示教器和上位机

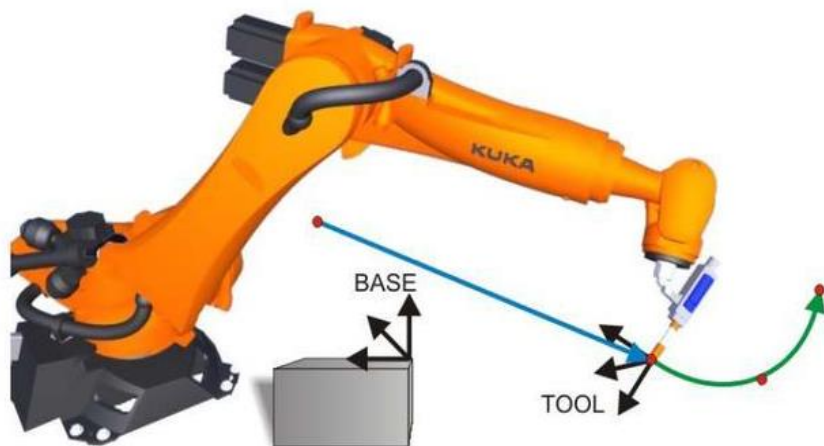


- 上位机——通常是工控机，内置开机启动的定制化机器人编程和操作程序
- 示教器——自带屏幕的小型定制计算机，供操作员手动控制和编程机器人

• 接收控制任务的过程

—— 从编程到上位机生成“路径指令”

发送至



```
MODULE M1
  CONST jointtarget A0:=[0,-25,15
  CONST robtarget A1:=[[-545.91,14
  CONST robtarget A11:=[[-545.91,1
  CONST robtarget A2:=[[-545.91,14
  PROC R1()
    MoveAbsJ A0\NoEOffs, v1000, z5
    MoveJ A1, v1000, z50, tool0\W0
    MoveL A2, v1000, z50, tool0\W
  .0
  .1  ENDPROC
```

2.2 机器人运动控制的实现

2.2.1 运动控制系统中的硬件——电机驱动

➤ 电机驱动

- 放大控制信号
- 输出大功率的驱动电压或电流给电机
- 电机产生驱动转矩
- 通过减速器或者直接拖动关节运转跟踪关节指令

来自



➤ 编码器

- 检测电机转角和速度
- 直接反馈给运动控制器
- 或者通过驱动器采集和变换再反馈

2.2 机器人运动控制的实现

2.2.2 运动控制器概述

- 运动控制器的功能
 - 可实现多个关节的协同控制，也被称为**多轴运动控制器**
 - 通过配置，可以实现**分散式或集中式**的控制方法
 - 高性能运动控制器的伺服环还能实现**力位混合控制**
- 运动控制器的形式
 - 基于DSP的高成本控制器 —— **高性能机械臂系统**
 - 基于ARM的低成本控制器 —— **低成本机械臂系统**
 - 基于Ethercat总线的软件 —— **组网要求高的中等性能机械臂系统**



(a) PCI总线型



(b) PC104总线型



(c) 独立型



(d) 现场总线型

几种典型的运动控制器

2.2 机器人运动控制的实现

2.2.2 运动控制器概述

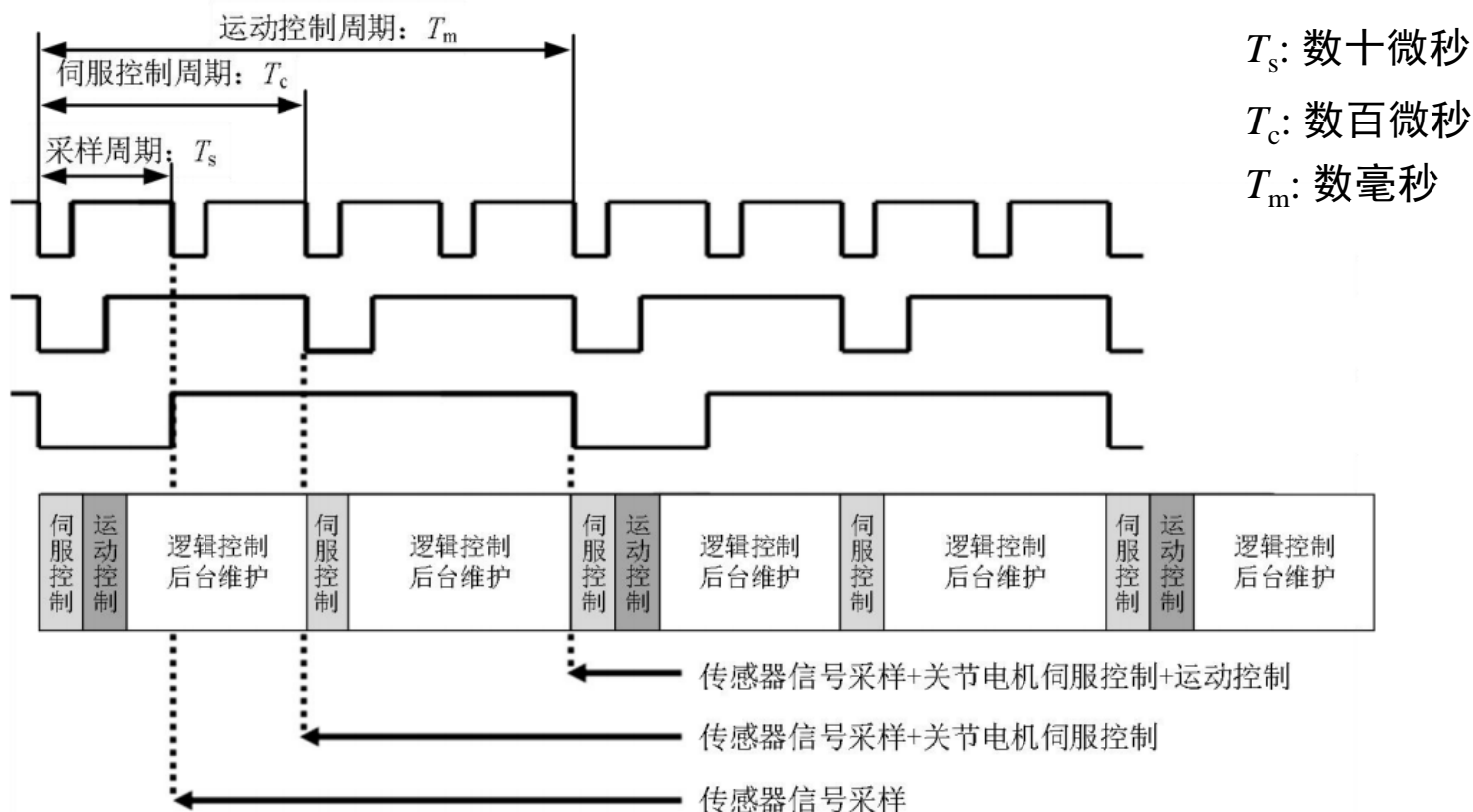
- 运动控制器的主要软件模块
 - 操作空间轨迹生成
 - 轨迹前瞻和速度调节
 - 运动学正/逆解
 - 各关节期望值更新
 - 各关节位置伺服/速度伺服
 - 多种传感器信号采样
 - 运动逻辑控制
 - 后台维护
 - 无刷电机的交流换向
 - 电流闭环控制

2.2 机器人运动控制的实现

2.2.2 运动控制器概述

- 内部时钟

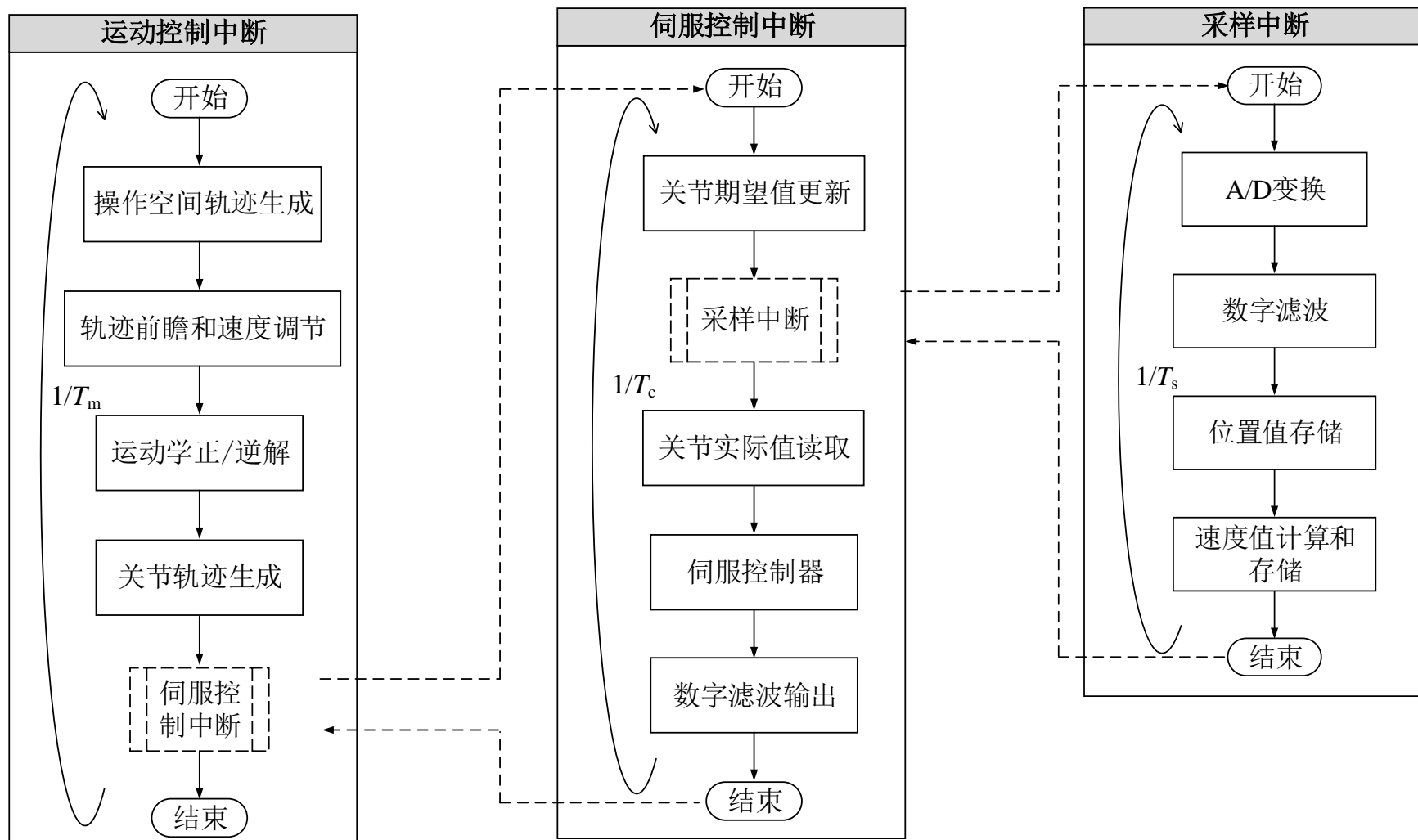
- 为保证实时性，运动控制器必须按固定周期运行不同的功能
- 在系统时钟周期的基础上进行分频，得到**采样周期**、**伺服周期**和**运动控制周期**



2.2 机器人运动控制的实现

2.2.2 运动控制器概述

● 内部时钟

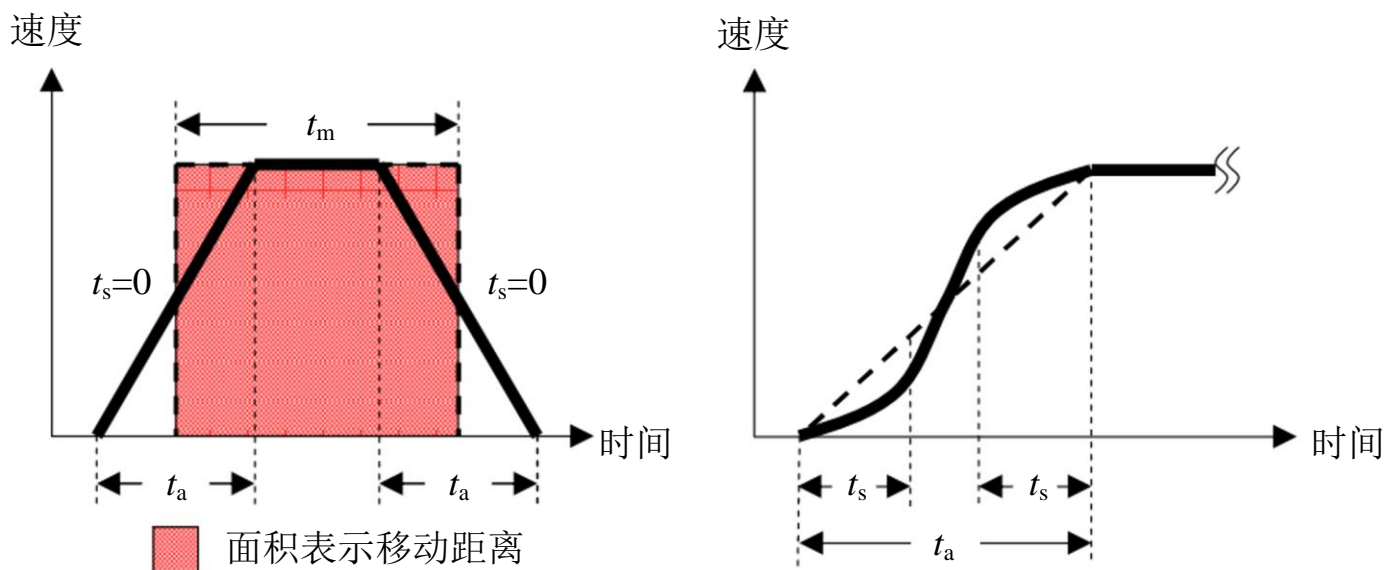


2.2 机器人运动控制的实现

2.2.3 运动控制器的软件模块

● 操作空间轨迹生成

- 在用户指定的路径点之间，用指令函数生成路径
- 常用的**路径指令函数**包括：**直线、圆弧、样条、位置-速度-时间（PVT）**等
- 路径指令既包含了空间信息，也包含了时间和速度信息
- 路径参数的时间函数一般采用速度S曲线



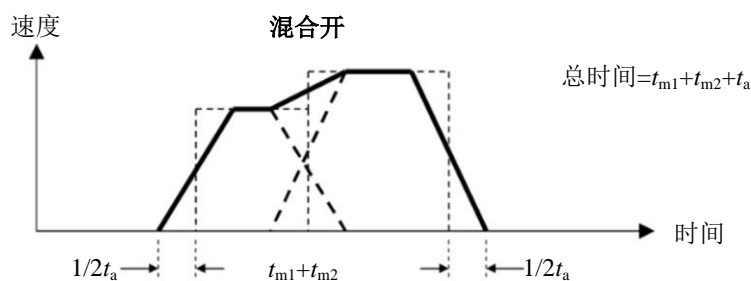
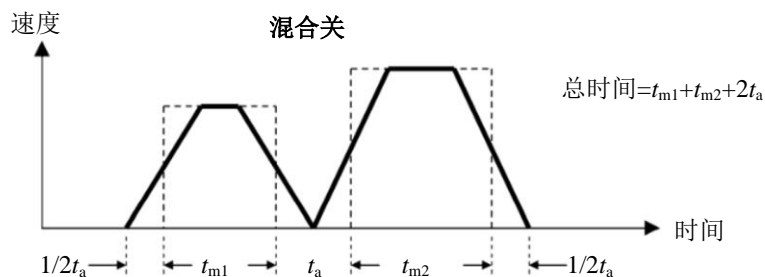
t_m : 参考时间 t_a : 加速时间 t_s : 变加速时间

2.2 机器人运动控制的实现

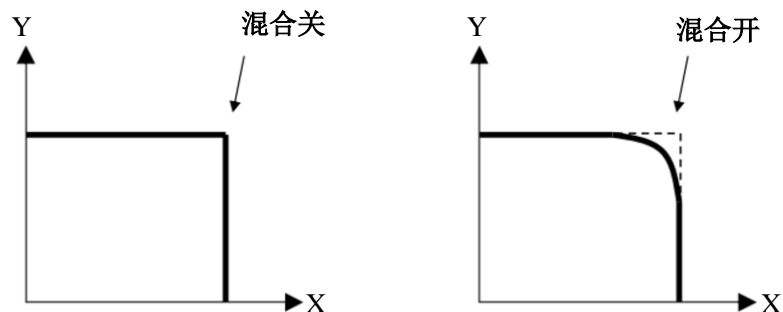
2.2.3 运动控制器的软件模块

● 操作空间轨迹生成

- 对于多条路径指令，运动控制器将进行速度混合，使速度连续，速度混合采用S曲线



速度混合



直线路径速度混合的效果

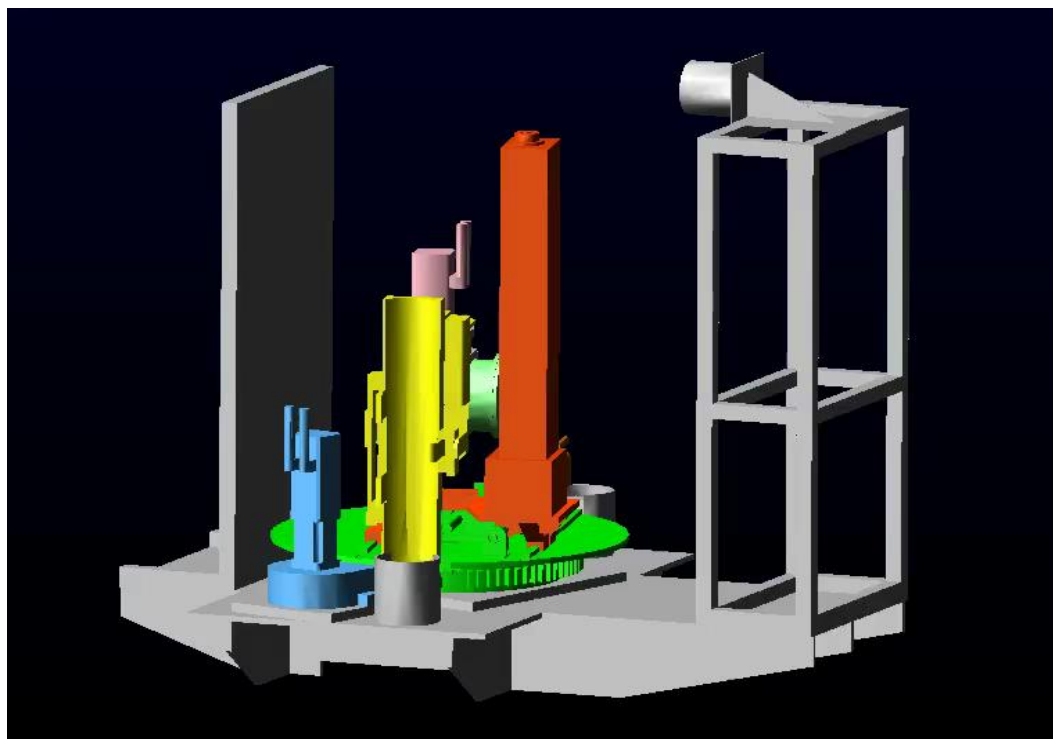
- 运动控制器在运动控制周期内完成操作空间轨迹生成

2.2 机器人运动控制的实现

2.2.3 运动控制器的软件模块

- 操作空间轨迹生成

- PVT模式：在路径点处的位移、速度、时间是指定值的约束条件下，以Hermite曲线（5次分段多项式）对路径点进行平滑连接，对复杂路径的描述更加精确。



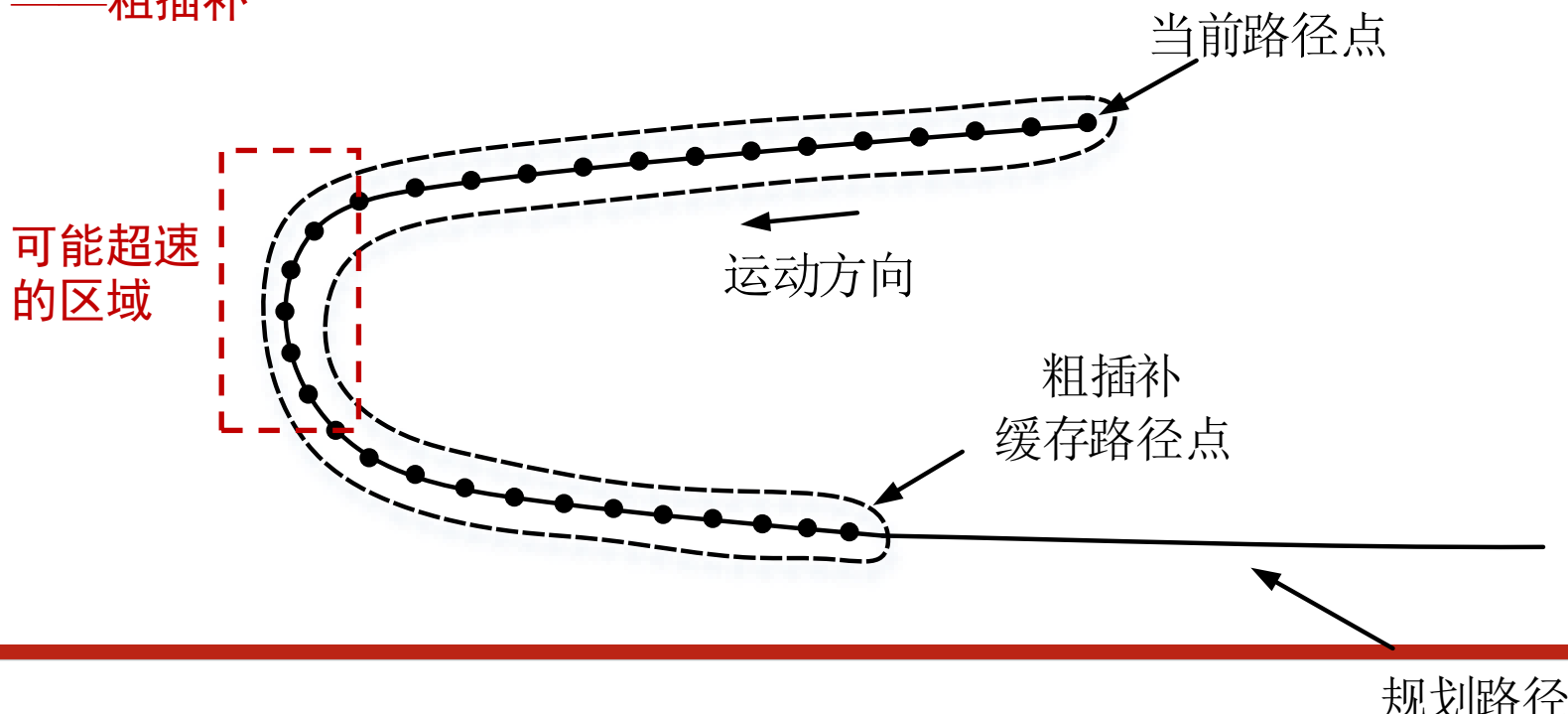
PVT模式生成的平滑空间轨迹

2.2 机器人运动控制的实现

2.2.3 运动控制器的软件模块

● 轨迹前瞻与速度调节

- 在笛卡尔空间设定的速度值，不能确保关节速度小于驱动电机速度上限，例如：奇异点附近或大曲率路径 —— **需要提前计算关节速度**
- 运行过程中可能存在速度变更，例如：避障或用户实时更新末端速度值 —— **需要轨迹前瞻**（在运行时提前预估是否可以按规定速度运行调节速度）
- 前瞻时，以5~20ms的实际运行时间间隔计算路径点位移和速度，并将其放入缓存 —— **粗插补**

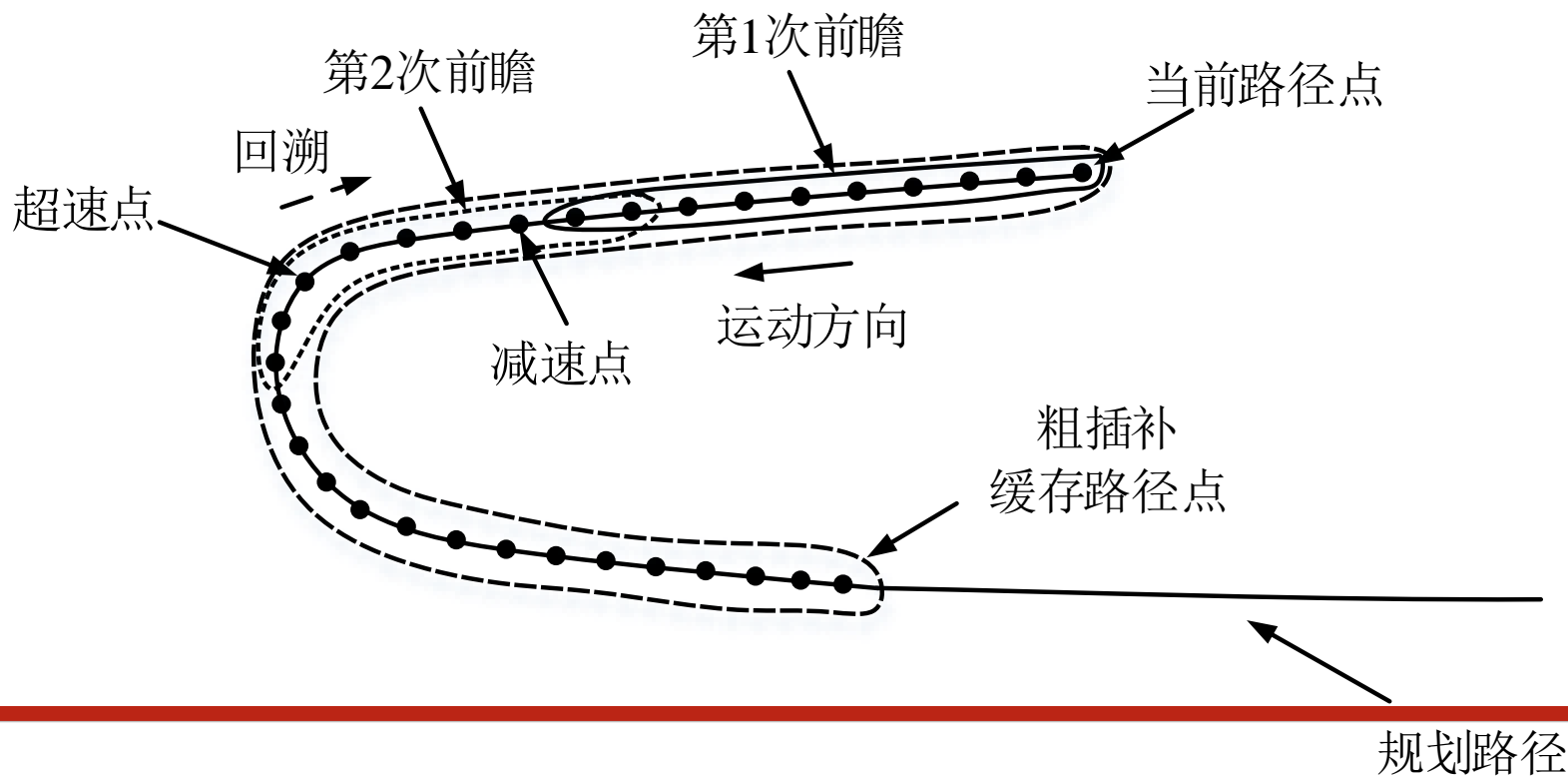


2.2 机器人运动控制的实现

2.2.3 运动控制器的软件模块

● 轨迹前瞻与速度调节

- 为了能够及时加减速，前瞻算法分段处理缓存数据，从当前路径点往前一段段对缓存中的路径点执行运动学逆解
- 如果检测到超速点（超过关节速度限定值），则回溯路径点
- 根据机器人最大减速能力找到减速点，从减速点开始降低沿路径的指令速度

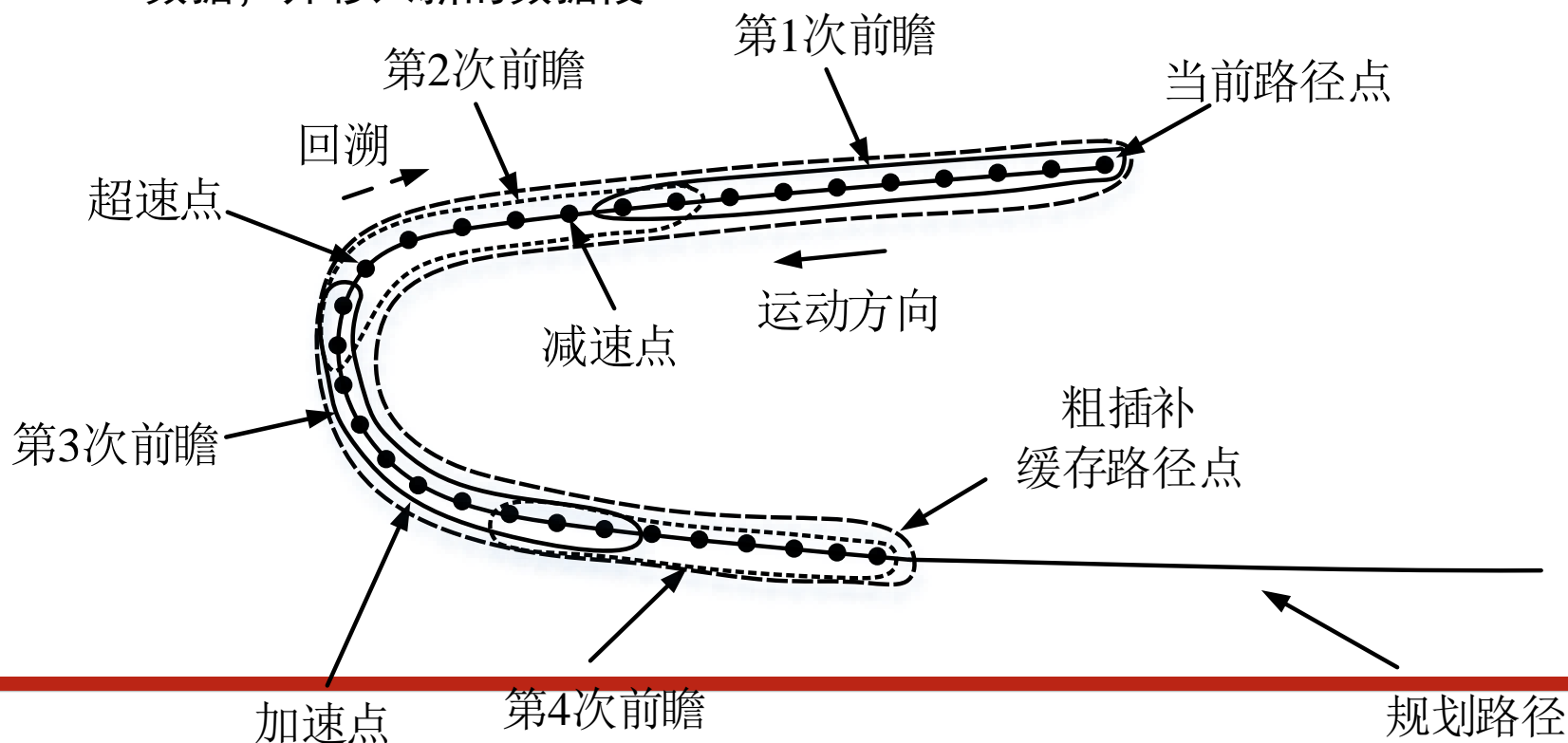


2.2 机器人运动控制的实现

2.2.3 运动控制器的软件模块

● 轨迹前瞻与速度调节

- 继续前瞻计算
- 如果检测到路径点速度低于用户指令速度，且关节速度也低于其最高限定值，则启动加速过程
- 软件按照先入先出（FIFO-First Input First Output）的原则，逐段移出已检查的数据，并移入新的数据段



2.2 机器人运动控制的实现

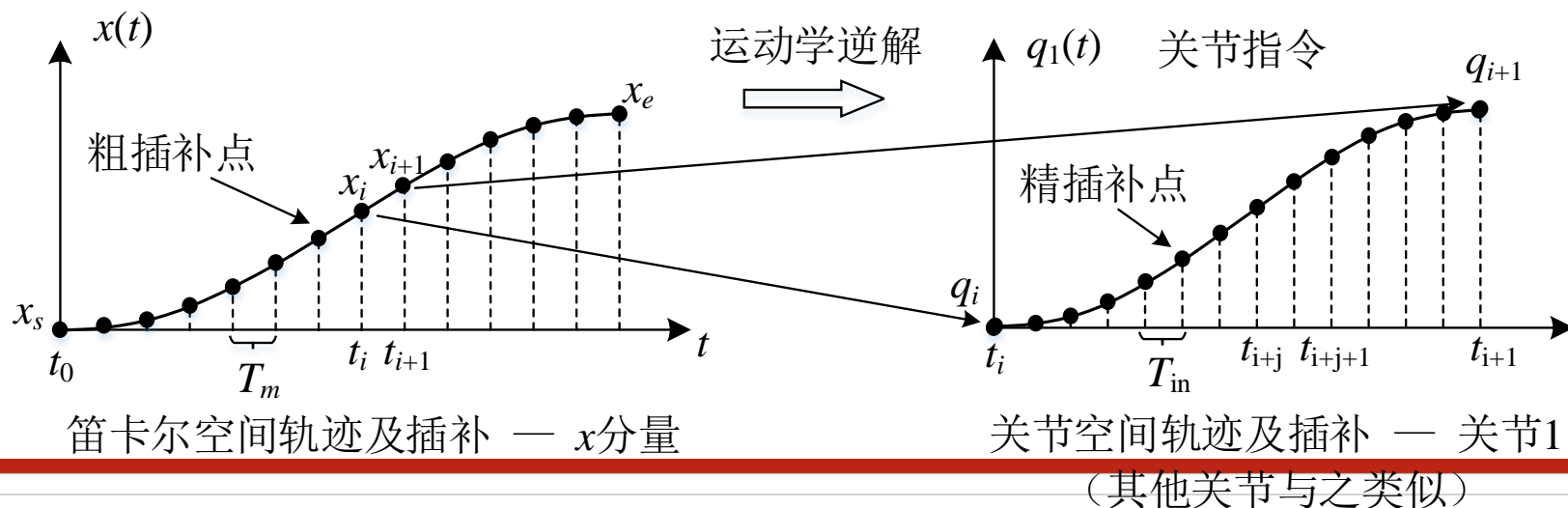
2.2.3 运动控制器的软件模块

- 运动学正/逆解

- 正解——根据关节位置计算末端位姿，使系统能够实时获取末端位姿
- 逆解——计算末端路径各粗插补点对应的关节位置，使伺服算法获得当前伺服控制指令，即**关节期望值**
- 理论上，每个伺服周期都要多次调用正/逆运动学方程

- 关节轨迹生成和关节期望值更新

- 在各粗插补点的关节位置之间，采用Hermite或样条函数生成关节轨迹
- 每3~5个伺服周期，在关节轨迹上进行**精插补**，得到当前时刻的关节期望值

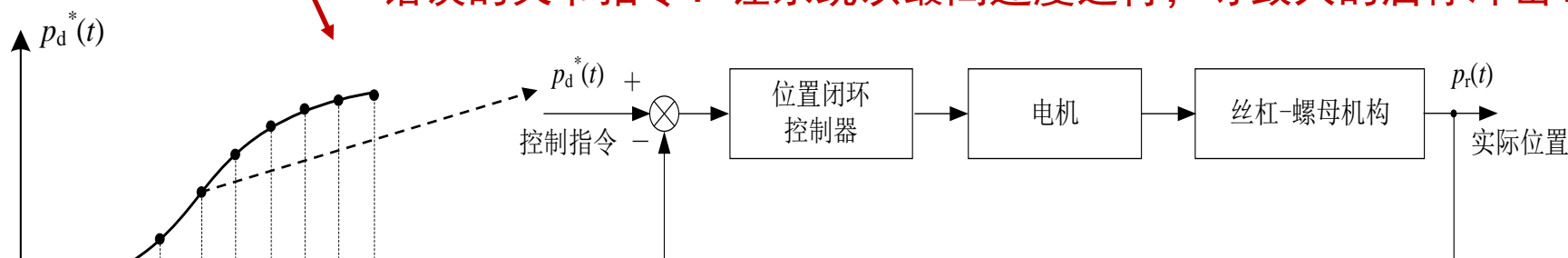
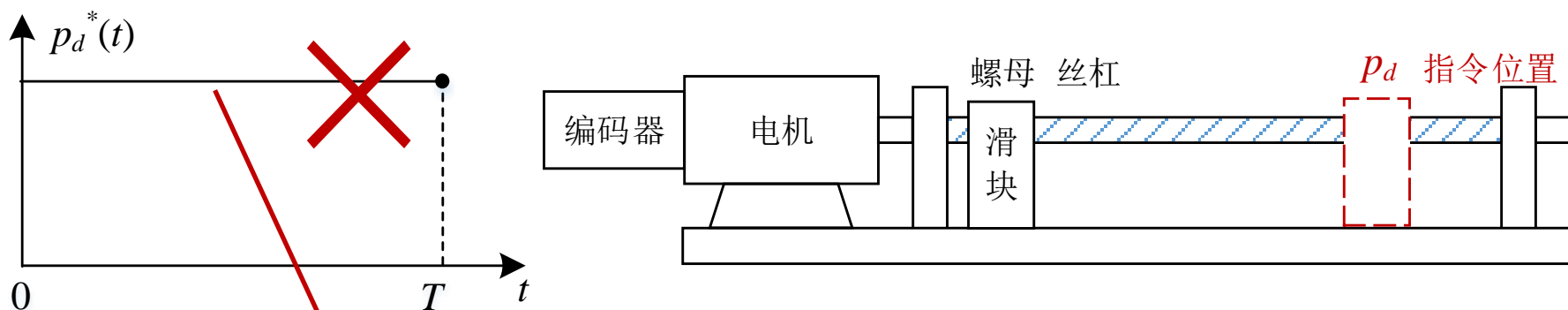


2.2 机器人运动控制的实现

2.2.3 运动控制器的软件模块

- 关节轨迹生成和关节期望值更新

- 对于一个数字伺服系统，关节轨迹生成过程是必需的
- 即使用户只给定了一个终点位置，也必须生成从当前点到终点的位置轨迹，然后插补生成控制指令，发送给伺服控制器，而不能直接把终点位置当作控制指令发给伺服控制器

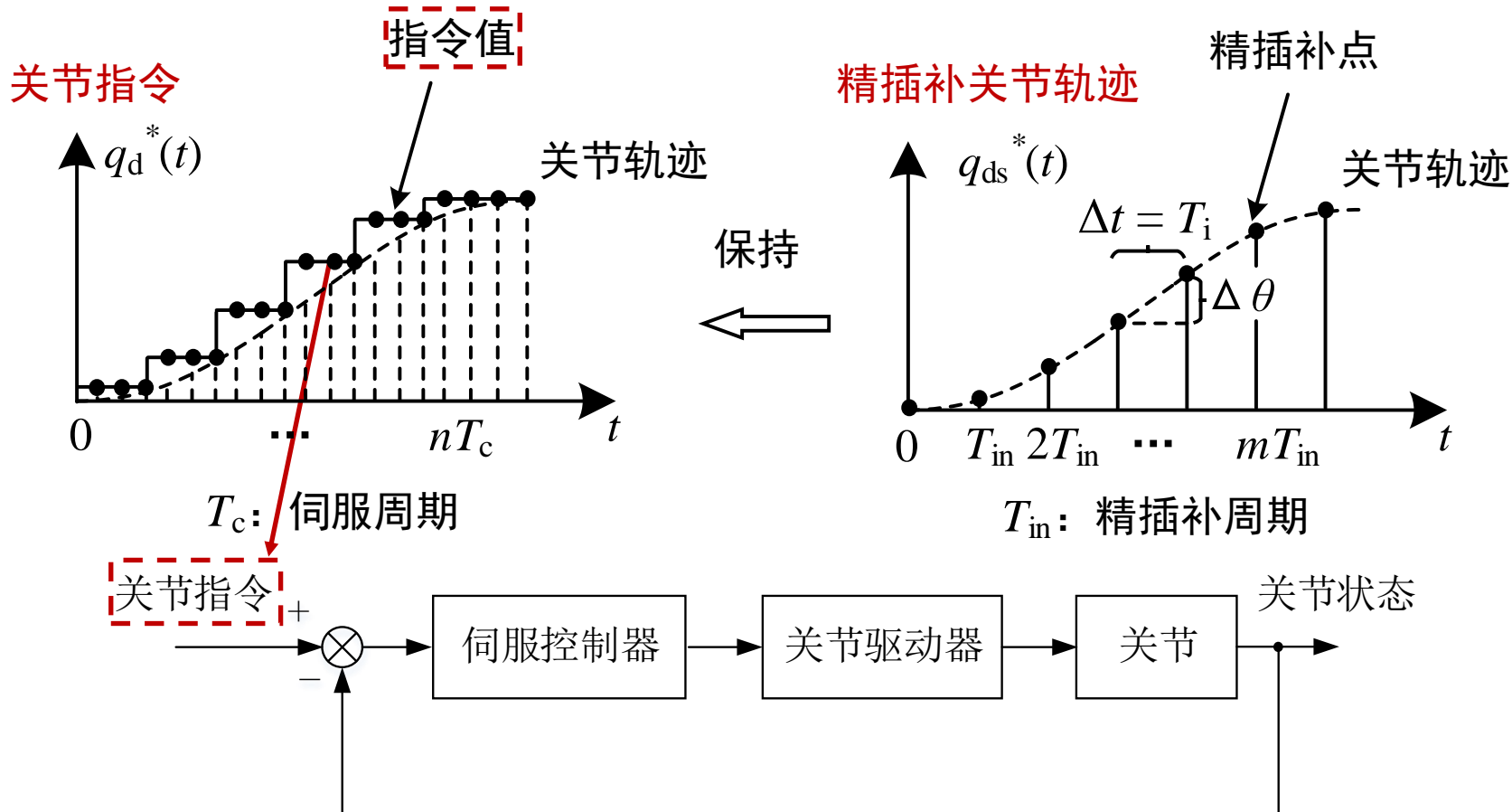


2.2 机器人运动控制的实现

2.2.3 运动控制器的软件模块

- 关节位置伺服/速度伺服

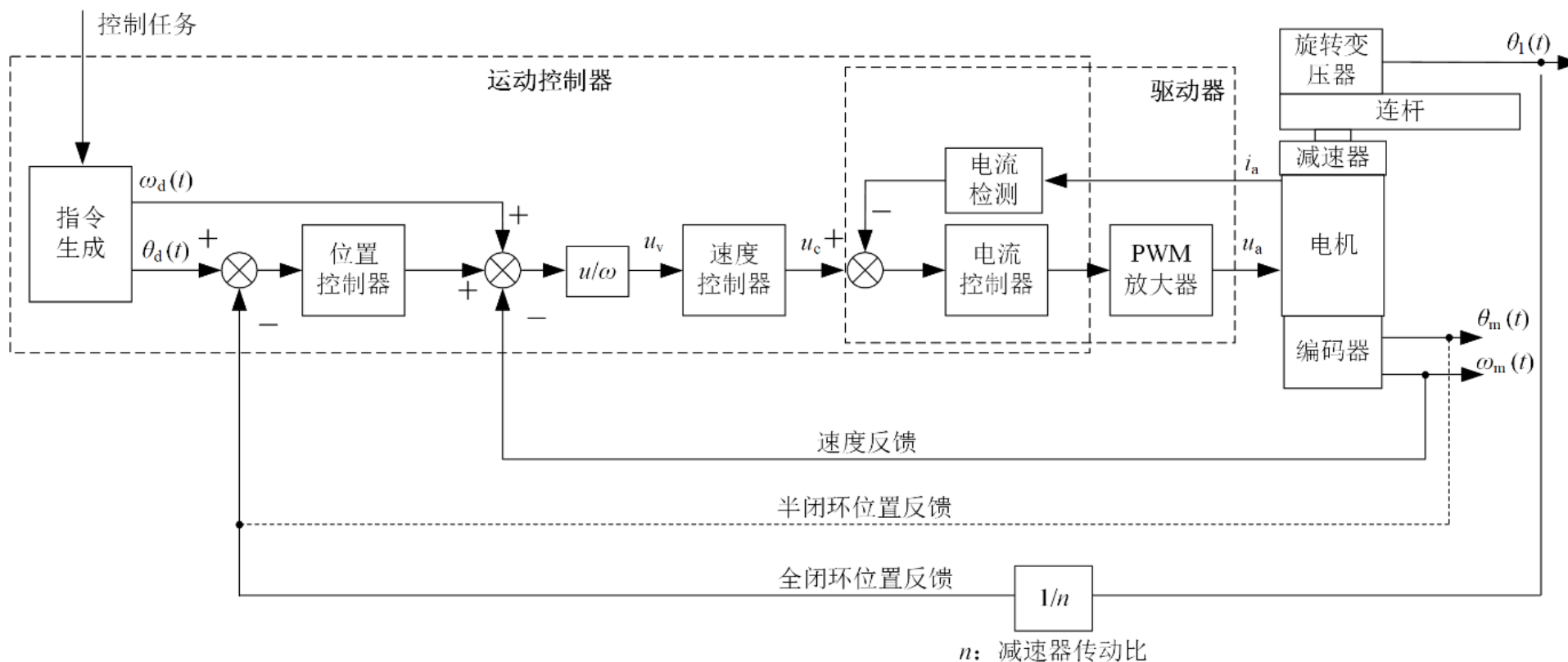
➤ **伺服控制器**是闭环控制算法和硬件的统称，它完成一次伺服计算并输出控制信号的时间就是**伺服周期**。



2.2 机器人运动控制的实现

2.2.3 运动控制器的软件模块

- 关节位置伺服/速度伺服



由运动控制器和驱动器构成典型三环位置控制系统

2.2 机器人运动控制的实现

2.2.3 运动控制器的软件模块

- 关节位置伺服/速度伺服

- PID控制算法

- 伺服环以含速度/加速度前馈的经典PID控制算法为默认控制器
- PID控制算法根据指令和偏差计算得到控制量，输出低功率控制信号 u_c ，经过电机驱动器转换成大功率信号 u_a

- 力矩模式和速度模式

- 对于工作在电流闭环状态下的电机及其驱动器，称其工作在**力矩模式**；否则，称其工作在**速度模式**

- 关节位置反馈

- 仅采用电机尾部的编码器，可构成“半闭环位置控制器系统”
- 采用关节位置传感器和编码器，可构成“全闭环位置控制系统”

- 电流闭环控制

- 替代电流伺服驱动器的电流闭环控制功能，使简单的驱动电路和电机的组合也能实现电流、速度、位置三闭环控制

2.3 机器人控制问题与控制器分类

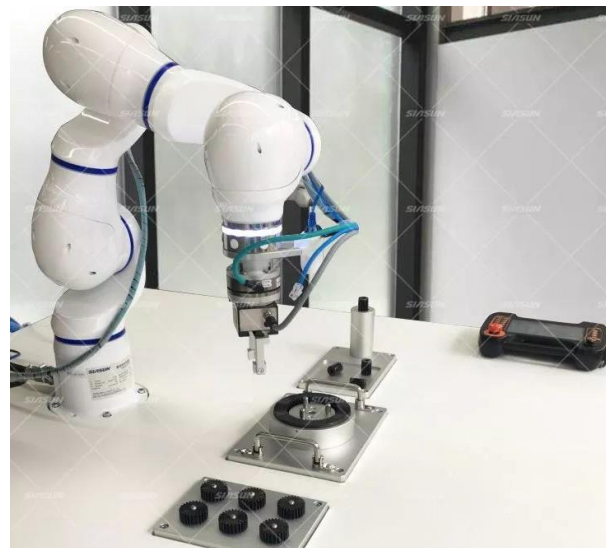
● 自由空间 —— $F_e = 0$

- 机械臂与环境**不发生接触**的工作模式，例如焊接、喷漆、涂胶等，是**运动控制问题**
- 只需考虑机器人自身的**动力学特性**，用**位置控制器**来解决



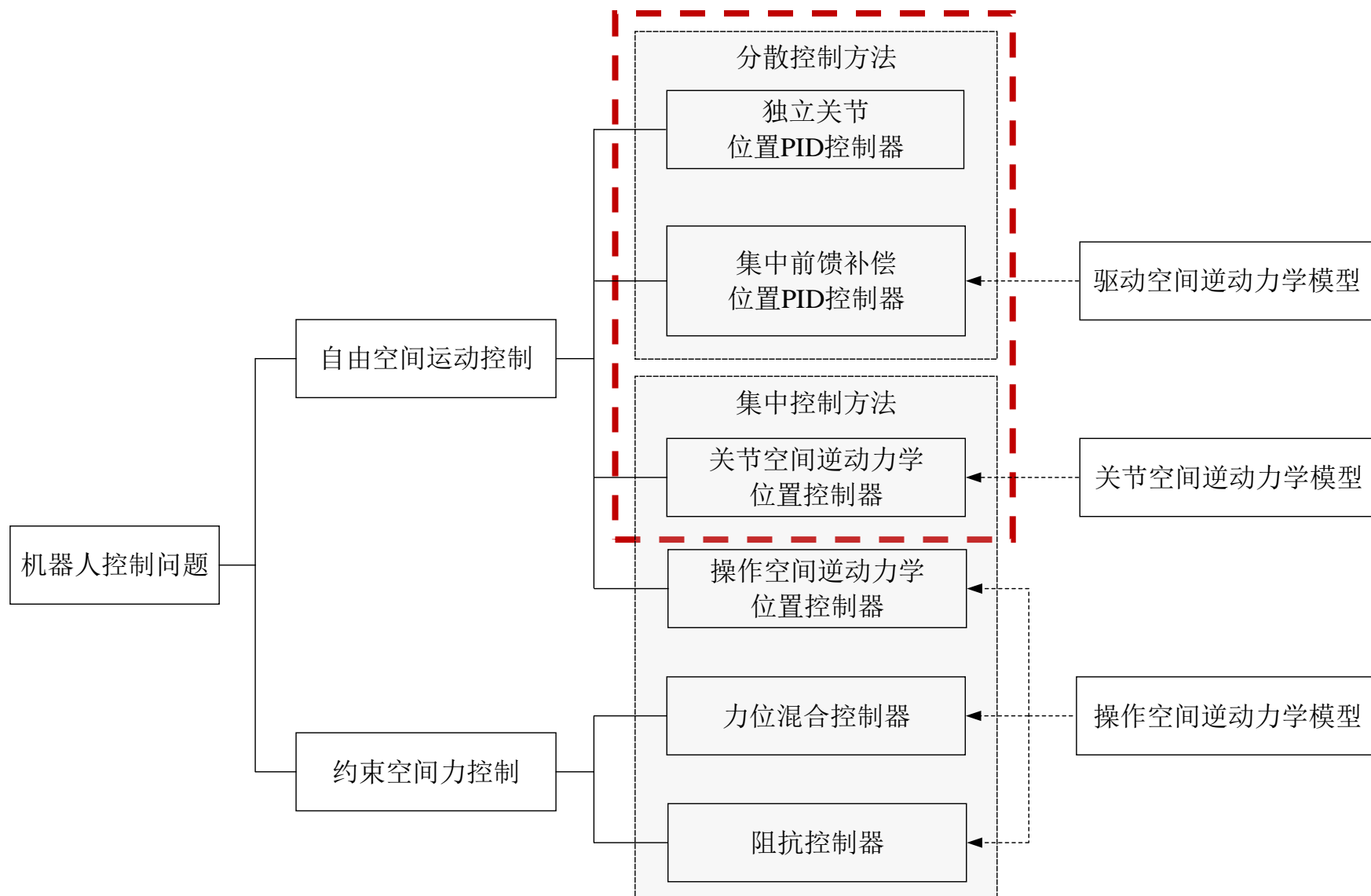
● 约束空间 —— $F_e \neq 0$

- 机械臂与环境**发生接触并有力作用**的工作模式，例如工件装配、人机协作等，涉及**力控制问题**
- 还需考虑**环境接触力**的影响，需要用**力位混合控制器**或**阻抗控制器**来解决



2.3 机器人控制问题与控制器分类

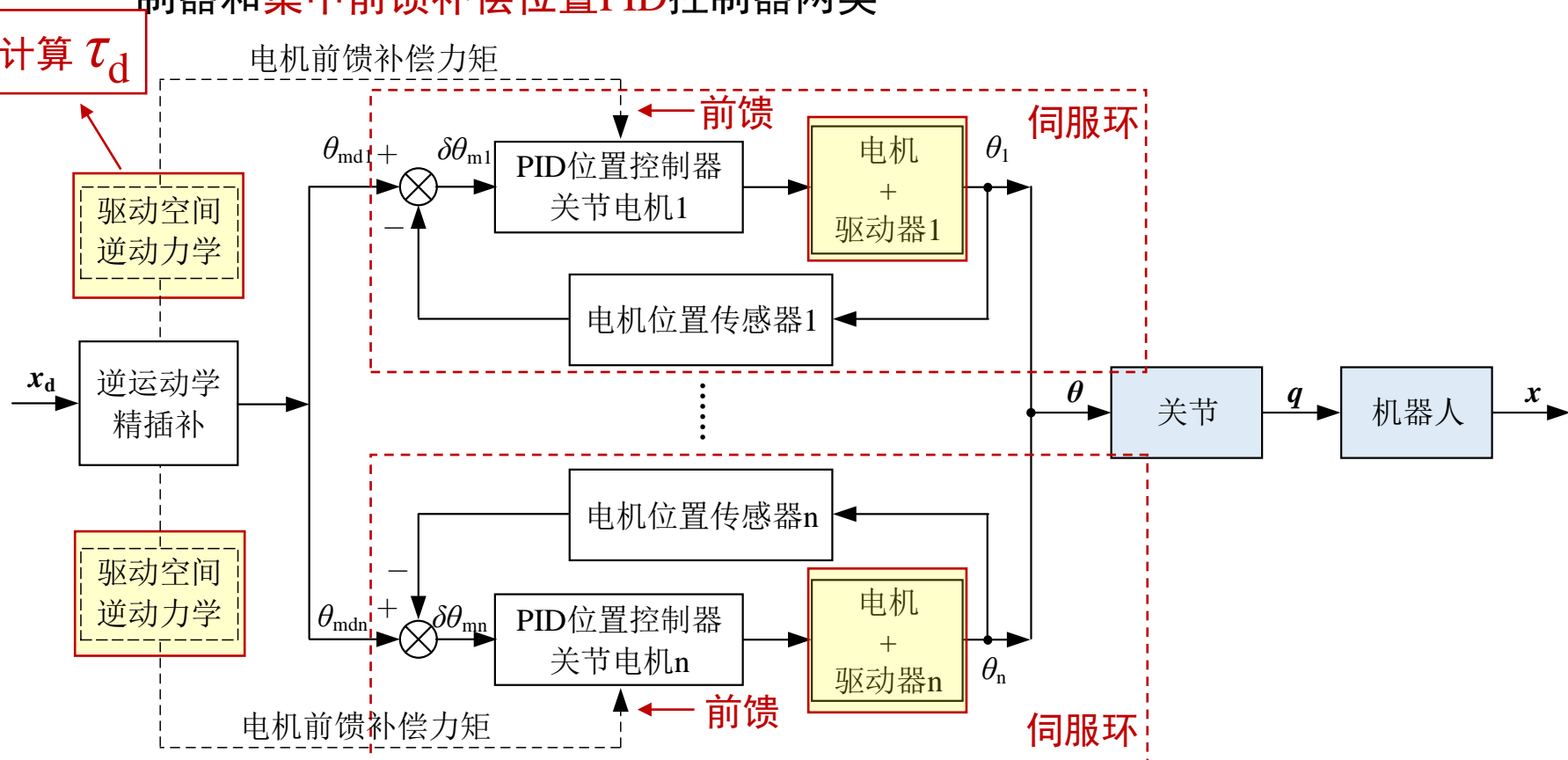
自由空间运动下的驱动/关节空间控制方案



2.3 机器人控制问题与控制器分类

- 驱动空间分散运动控制方法原理

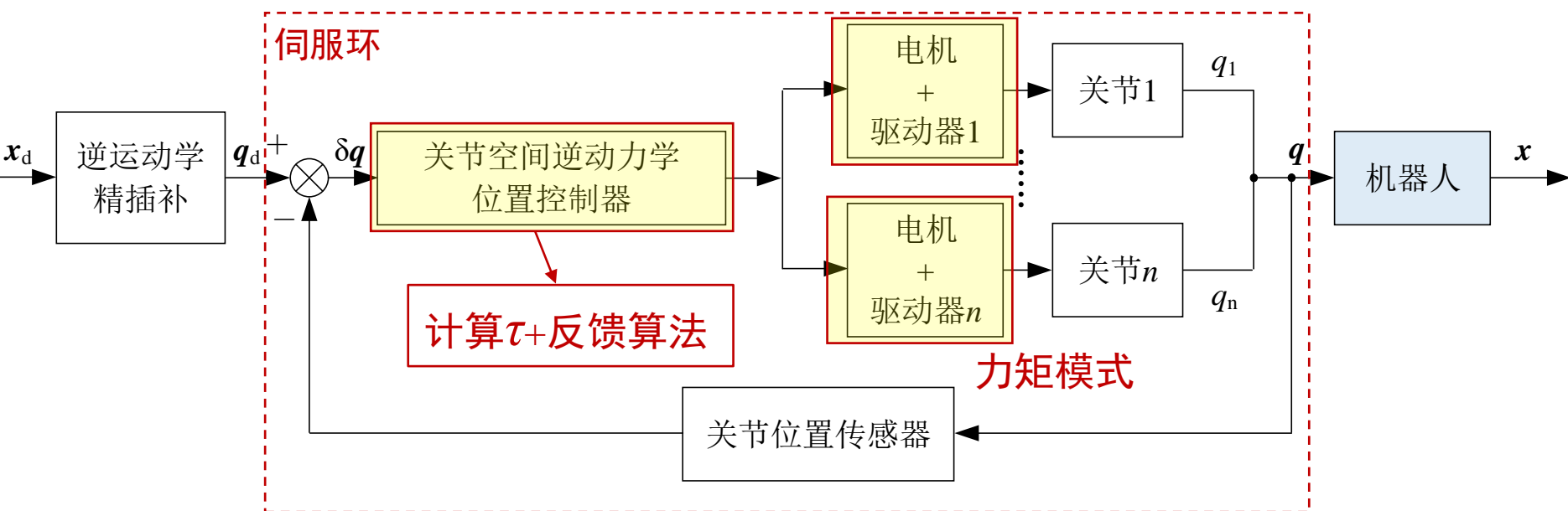
- 每个关节电机都由一个位置控制器（低成本单片机）独立控制
- 不考虑系统动力学，仅依据**驱动器**和**电机模型**设计控制参数
- 根据是否采用关节集中前馈补偿，分散控制方法分为**独立关节位置PID**控制器和**集中前馈补偿位置PID**控制器两类



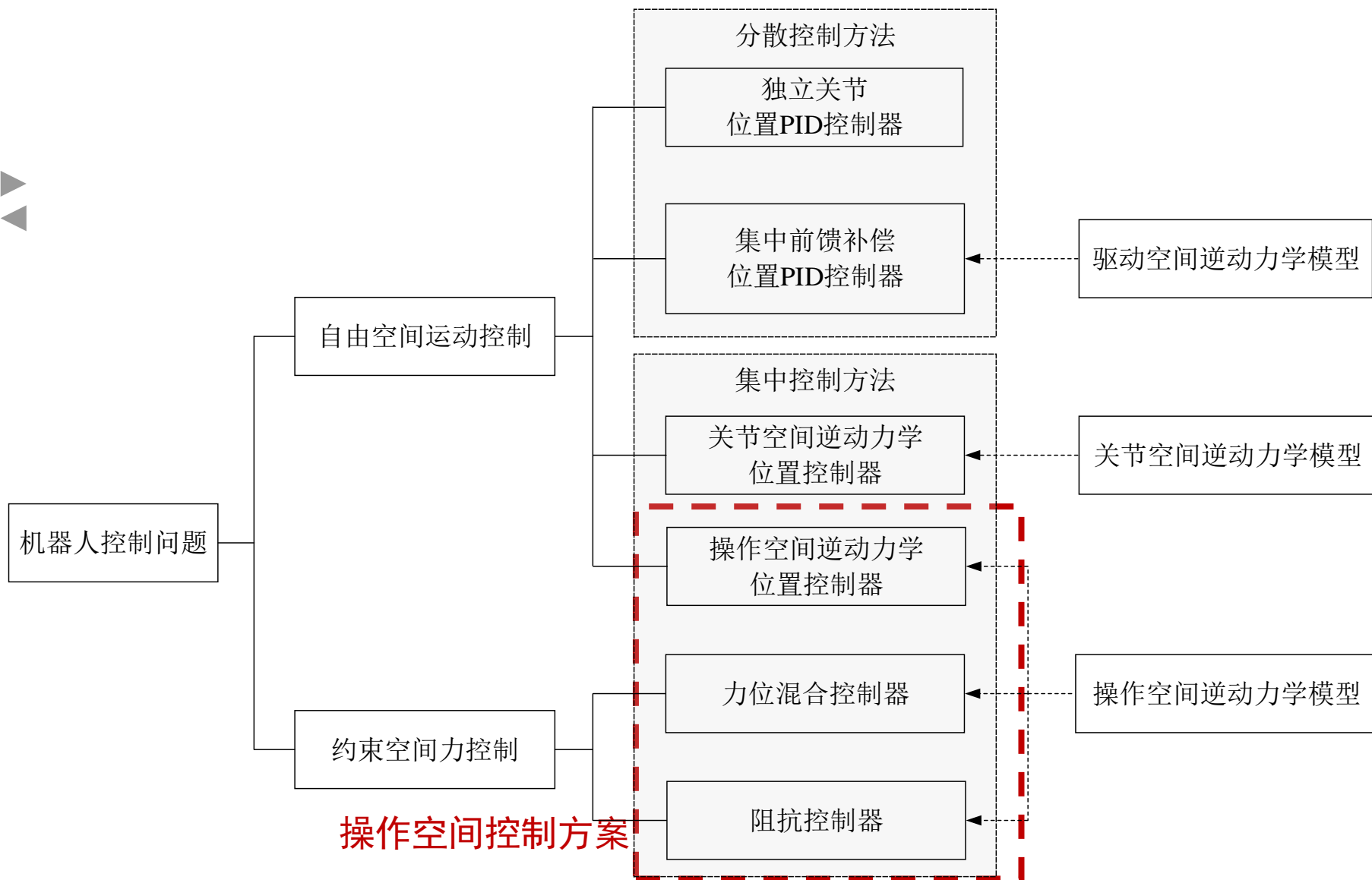
2.3 机器人控制问题与控制器分类

● 关节空间集中运动控制方法

- 各关节运动由统一的伺服控制器控制，属于集中控制架构
- 伺服环内部依据关节逆动力学模型计算各关节控制力矩
- 可有效补偿重力、关节耦合力等非线性因素
- 根据状态误差对状态微分项进行修正
- 对控制器计算性能要求高，要求电机及驱动器支持力矩模式



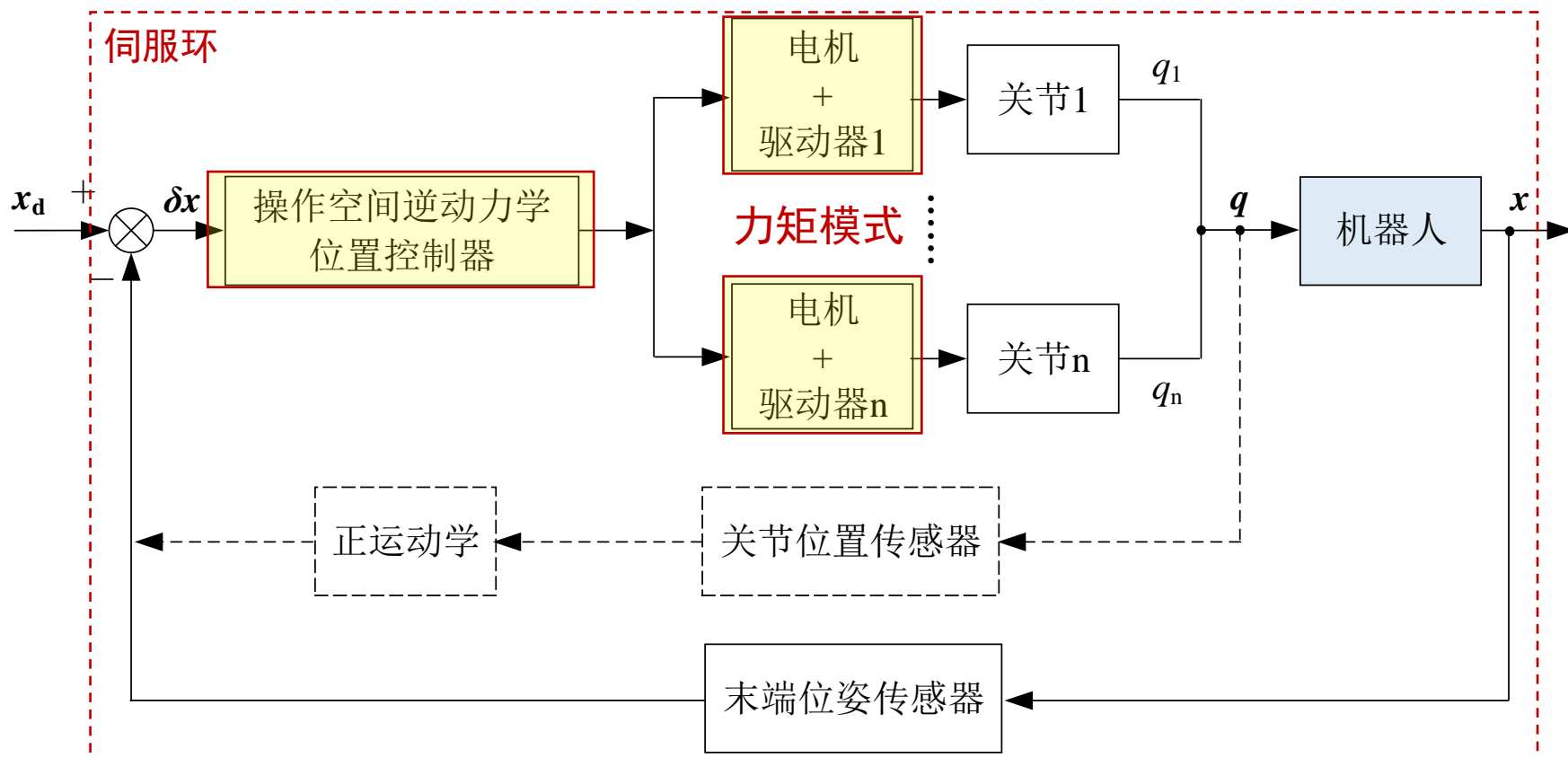
2.3 机器人控制问题与控制器分类



2.3 机器人控制问题与控制器分类

● 操作空间集中运动控制方法

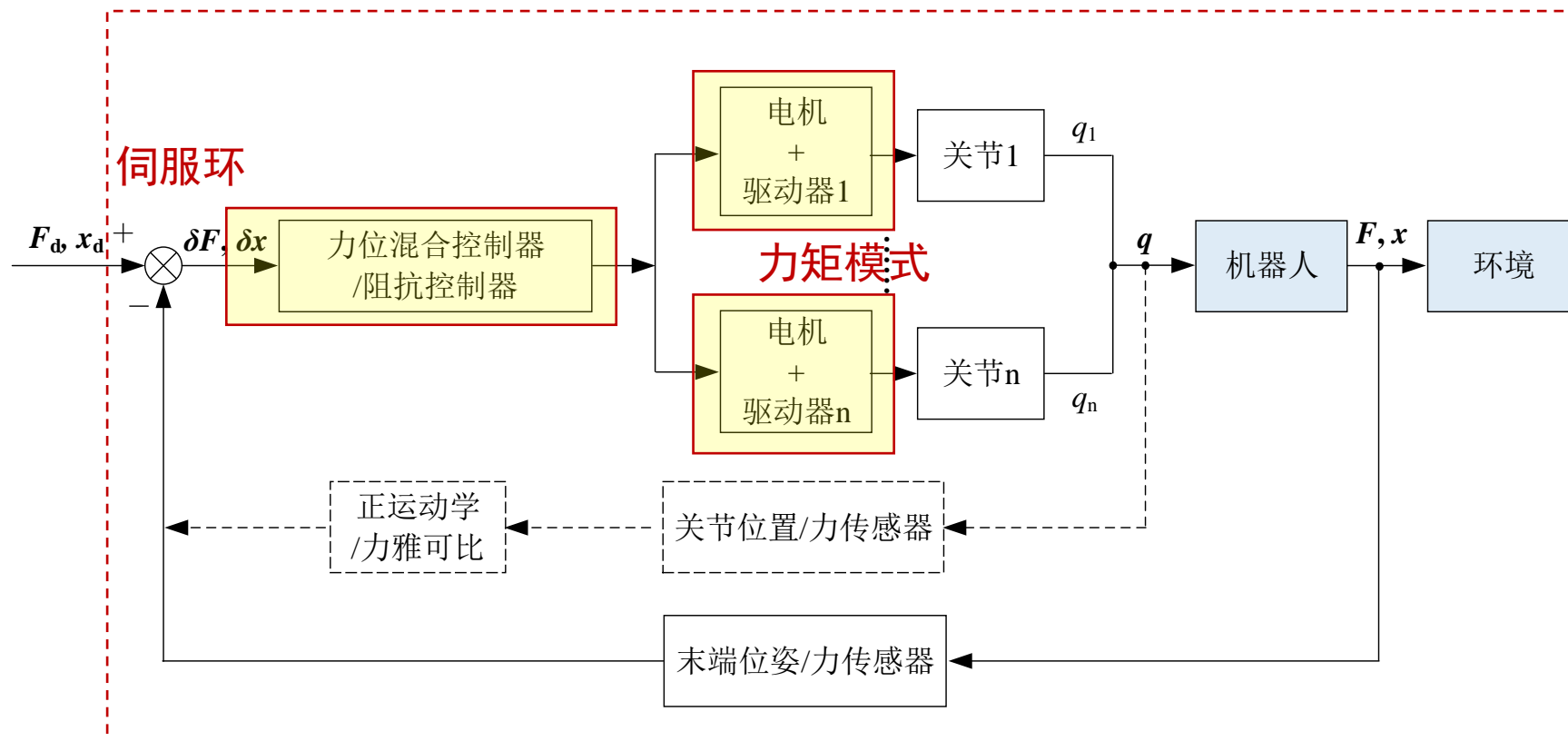
- 集中式伺服控制器在伺服环内进行操作空间逆动力学计算，实施末端位姿闭环控制，**根据所需广义力求解关节控制力**
- 对控制器计算性能要求高，要求电机及驱动器支持**力矩模式**
- 工程中常根据**关节测量值**，根据正运动学模型计算末端位姿作为反馈



2.3 机器人控制问题与控制器分类

● 操作空间集中式力控制方法

- 面向机器人与环境发生力交互时，如何保证运动精度或/和交互力精度
- 关注操作空间中末端执行器的位移和接触力，利用**操作空间逆动力学**方程计算关节力矩



研究内容

查阅资料，写一篇EtherCAT现场总线在工业机器人运动控制中的应用方面的小论文。

课后作业

作业

- 1、简述路径与轨迹的定义。
- 2、路径规划与轨迹生成的主要区别是什么？
- 3、示教编程和离线编程各自的特点是什么？需要什么设备支持编程？
- 4、笛卡尔空间轨迹生成与关节空间轨迹生成的区别是什么？
- 5、为什么笛卡尔空间轨迹生成之后要进行粗插补？粗插补的结果是什么？
- 6、为什么关节空间轨迹生成之后要进行精插补？其结果是什么？在关节闭环控制中起到什么作用？
- 7、为什么粗插补周期比精插补周期长？
- 8、对于实现位置闭环的数字控制系统，为什么要针对指令位置进行轨迹生成？简述根据指令位置生成位置轨迹，并得到位置伺服指令值得过程。
- 9、运动控制器的主要功能是什么？
- 10、运动控制器的三个重要中断软件模块是什么？简述它们之间的时序关系。
- 11、为什么要进行轨迹前瞻和速度调节？