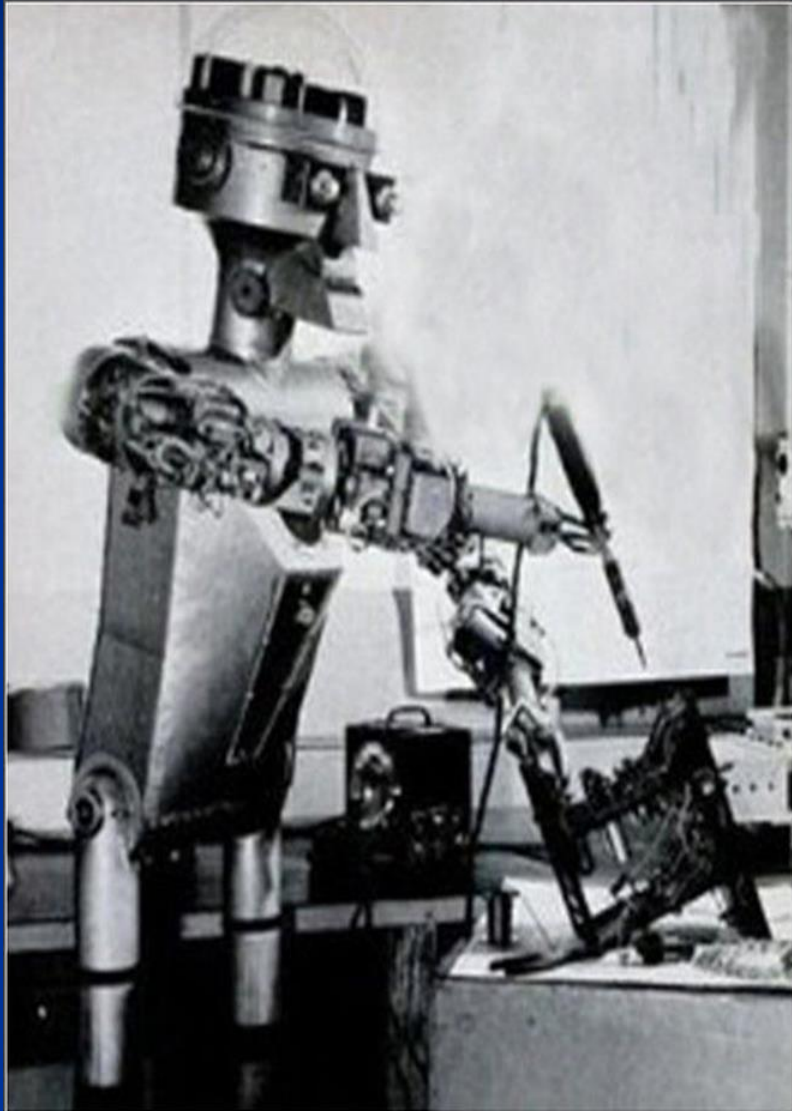


南昌理工学院

Nanchang Institute of Technology



工业机器人

机电工程学院

第七章 工业机器人轨迹规划与编程

7.1 工业机器人轨迹规划

7.2 关节空间法

7.3 直角坐标空间法（自学）

7.4 轨迹的实时生成（自学）

7.5 工业机器人编程

7.6 工业机器人编程语言（自学）

7.7 工业机器人离线编程

本章主要内容：

◆ 轨迹规划

◆ 机器人语言

◆ 机器人离线编程

机器人离线编程系统是利用计算机图形学，建立机器人编程环境，从而可以脱离机器人工作现场进行编程的系统。由于不占用机动时间，提高了设备利用率。而且由于离线编程本身就是CAD/CAM一体化的组成部分，有时可以直接利用CAD数据库的信息，大大减少了编程时间，提高了编程水平。

7.1 工业机器人的轨迹规划

一、引言

指定工业机器人执行某项操作作业

附加一些约束条件

轨迹的规划和协调

关节坐标空间

直角坐标空间

引言

轨迹规划涉及下面三个问题：

(1) 要对机器人的任务进行描述，即对机器人的运动轨迹进行描述。

(2) 根据所确定的轨迹参数，如何在计算机内部描述所要求的轨迹。这主要是选择习惯规定以及合理的软件数据结构问题。

轨迹规划器具有简化任务描述的功能，用户不需安

(3) 对内部描述的轨迹进行实际计算。通常是在运行时间内按一定的速率计算出位置、速度和加速度，生成运动轨迹。

划的细节问题则由示教单元来完成。

二、轨迹规划的一般性问题

工业机器人的作业可以描述成工具坐标系 $\{T\}$ 相对于工作坐标系 $\{S\}$ 的一系列运动：

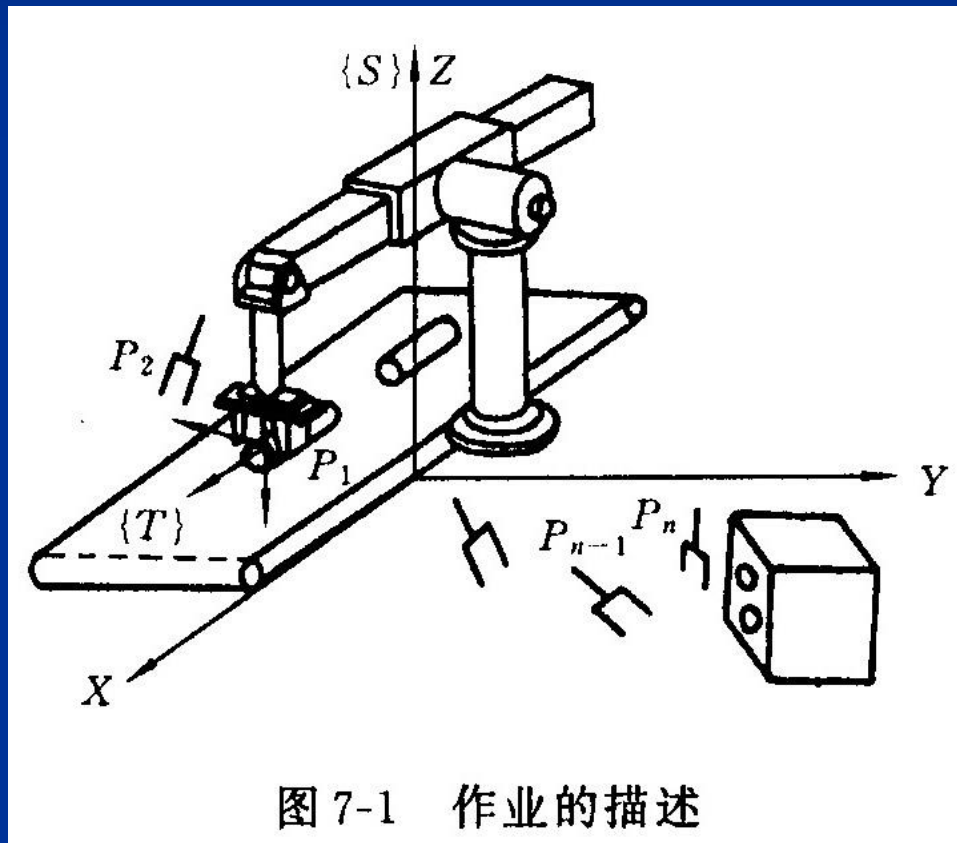
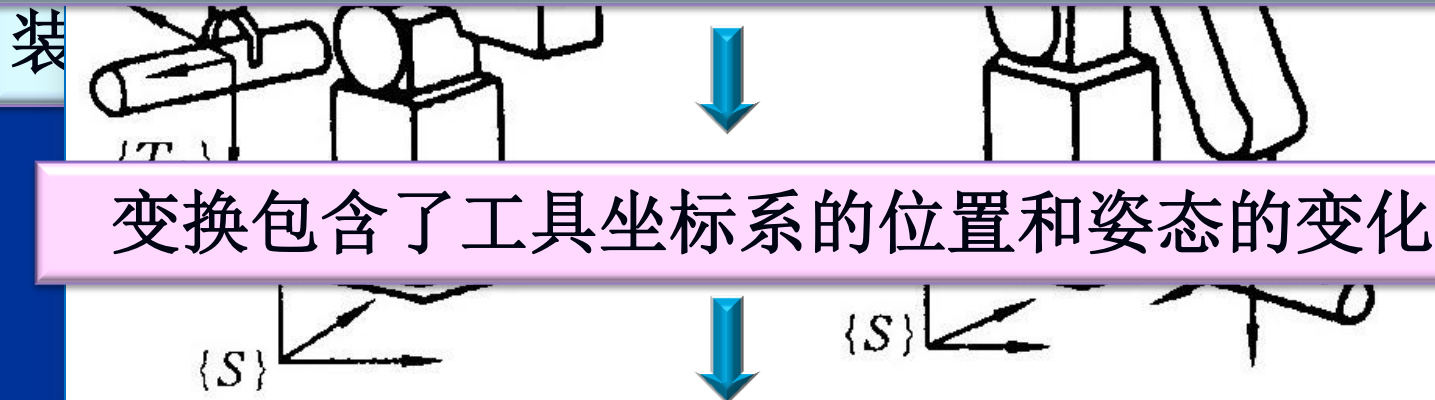


图 7-1 作业的描述

轨迹规划的一般性问题

工具坐标系相对于工作坐标系的运动来描述作业路径。

机器人从初始状态运动到终止状态的作业，看成是工具坐标系从初始位置 $\{T_0\}$ 变化到终止位置 $\{T_f\}$ 的坐标变换。



变换包含了工具坐标系的位置和姿态的变化。

在轨迹规划中，也常用“点”这个词来表示机器人的状态，或用它来表示工具坐标系的位姿。

轨迹规划的一般性问题

当需要更详细地描述运动时，不仅要规定机器人的起始点和终止点，而且要给出介于起始点和终止点之间的中间点，也称**路径点**。

运动轨迹的起始点和终止点之间必须给出两个路径点之间的时间分配。一阶导数(速度)，有时甚至二阶导数(加速度)也应该连续。

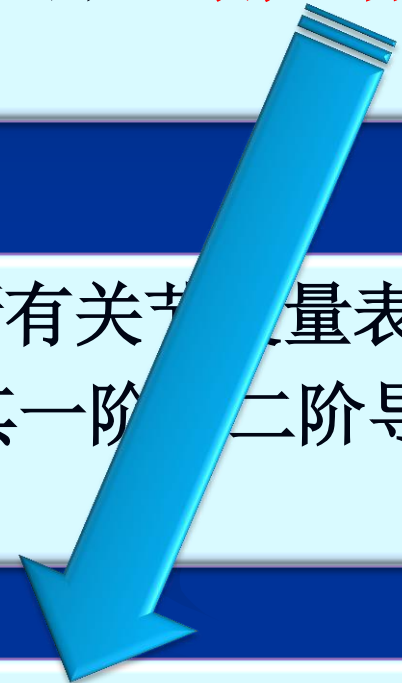
机器人的**运动应当平稳**，不平稳的运动将加剧机械部件的磨损，并导致机器人的振动和冲击。

轨迹规划的一般性问题

轨迹规划既可在**关节空间**中进行，也可在**直角坐标空间**中进行。



在关节空间中进行轨迹规划是指将所有关节位置量表示为时间的函数，用这些关节函数及其一阶、二阶导数描述机器人预期的运动。



在直角坐标空间中进行轨迹规划，是指将手爪位姿、速度和加速度表示为时间的函数，而相应的关节位置、速度和加速度由手爪信息导出。

第七章 工业机器人轨迹规划与编程

7.1 工业机器人轨迹规划

7.2 关节空间轨迹规划

7.3 直角坐标空间法（自学）

7.4 轨迹的实时生成（自学）

7.5 工业机器人编程

7.6 工业机器人编程语言（自学）

7.7 工业机器人离线编程

7.2 关节空间的轨迹规划

机器人作业路径点通常由工具坐标系 $\{T\}$ 相对于工作坐标系 $\{S\}$ 的位姿来表示，因此，在关节空间中进行轨迹规划：

◆ 首先需要将每个作业路径点向关节空间变换，即用逆运动学方法把路径点转换成关节角度值，或称关节路径点；

◆ 然后，为每个关节相应的关节路径点拟合光滑函数；

这些关节函数分别描述了机器人各关节从起始点开始，依次通过路径点，最后到达某目标点的运动轨迹。由于每个关节在相应路径段运行的时间相同，这样就保证了所有关节都将同时到达路径点和目标点，从而也保证了工具坐标系在各路径点具有预期的位姿；

关节空间的轨迹规划

◆ 关节空间中进行轨迹规划，不需考虑直角坐标空间中两个路径点之间的**轨迹形状**，仅以关节角度的函数来描述机器人的轨迹，计算简单、省时；

◆ 关节空间与直角坐标空间并不是连续的对应关系，关节空间内不会发生机构的**奇异现象**，从而避免了在直角坐标空间规划时所出现的关节速度失控问题；

在关节空间进行轨迹规划，**规划路径不是唯一的**。只要满足路径点上的约束条件，可以选取不同类型的关节角度函数，生成不同的轨迹。

1. 三次多项式插值

当已知末端操作器的起始位姿和终止位姿时，由逆向运动学，即可求出对应于两位姿的各个关节角度。因此，末端操作器实现两位姿的运动轨迹描述，可在关节空间中用通过起始点关节角和终止点关节角的一个平滑轨迹函数 $\theta(t)$ 来表示；

为了实现关节的平稳运动，每个关节的轨迹函数 $\theta(t)$ 至少需要满足四个约束条件：**两端点位置约束**和**两端点速度约束**。

三次多项式插值

端点位置约束是指起始位姿和终止位姿分别所对应的关节角度；

$$\begin{cases} \theta(0) = \theta_0 \\ \theta(t_f) = \theta_f \end{cases}$$

为满足关节运动速度的连续性要求，在起始点和终止点的关节速度可简单地设定为零，即；

$$\begin{cases} \dot{\theta}(0) = 0 \\ \dot{\theta}(t_f) = 0 \end{cases}$$

三次多项式插值

上述给出的四个约束条件可以唯一地确定一个三次多项式

$$\theta(t) = a_0 + a_1t + a_2t^2 + a_3t^3$$

运动过程中的关节速度和加速度则为：

$$\begin{cases} \dot{\theta}(t) = a_1 + 2a_2t + 3a_3t^2 \\ \ddot{\theta}(t) = 2a_2 + 6a_3t \end{cases}$$

三次多项式插值

为了求得三次多项式的系数，代以给定的约束条件，有方程组

$$\begin{cases} \theta_0 = a_0 \\ \theta_f = a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 \\ 0 = a_1 \\ 0 = a_1 + 2a_2 t_f + 3a_3 t_f^2 \end{cases}$$

三次多项式插值

求解上述方程组，可得

$$\begin{cases} a_0 = \theta_0 \\ a_1 = 0 \\ a_2 = \frac{3}{t_f^2} (\theta_f - \theta_0) \\ a_3 = -\frac{2}{t_f^3} (\theta_f - \theta_0) \end{cases}$$

三次多项式插值

所以，对于初速及终速为零的关节运动，满足连续平稳运动要求的三次多项式插值函数为

$$\theta(t) = \theta_0 + \frac{3}{t_f^2}(\theta_f - \theta_0)t^2 - \frac{2}{t_f^3}(\theta_f - \theta_0)t^3$$

其关节角速度和角加速度表达式为

$$\dot{\theta}(t) = \frac{6}{t_f^2}(\theta_f - \theta_0)t - \frac{6}{t_f^3}(\theta_f - \theta_0)t^2$$

$$\ddot{\theta}(t) = \frac{6}{t_f^2}(\theta_f - \theta_0) - \frac{12}{t_f^3}(\theta_f - \theta_0)t$$

三次多项式插值

三次多项式插值的运动轨迹曲线

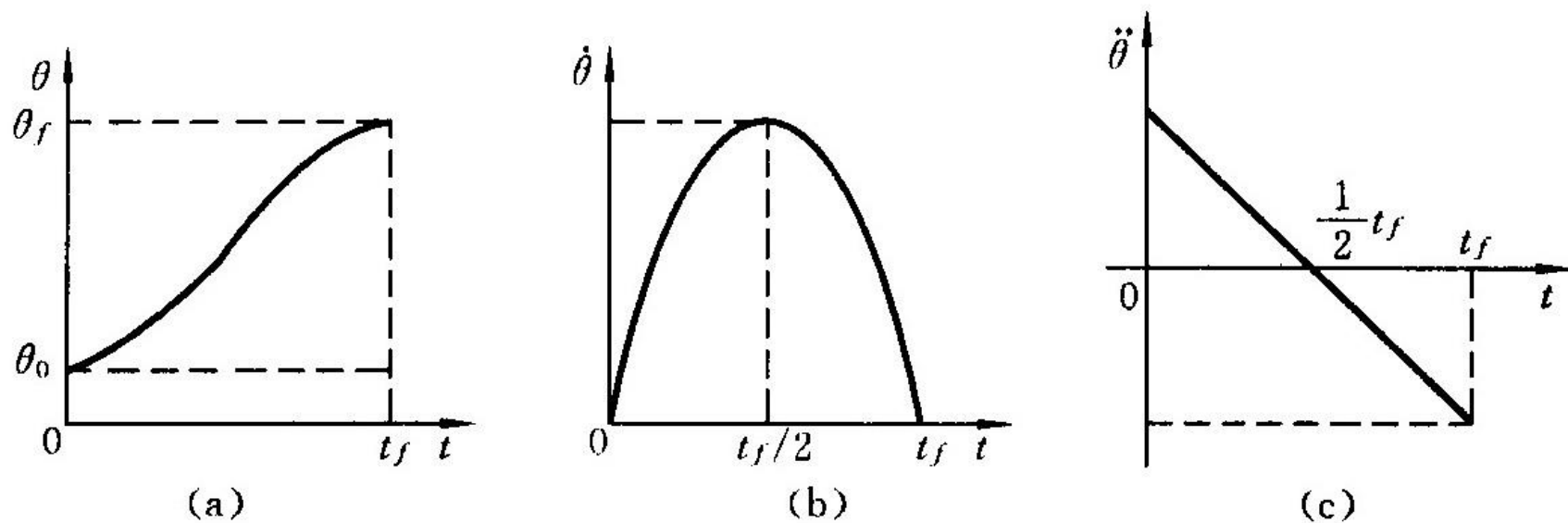
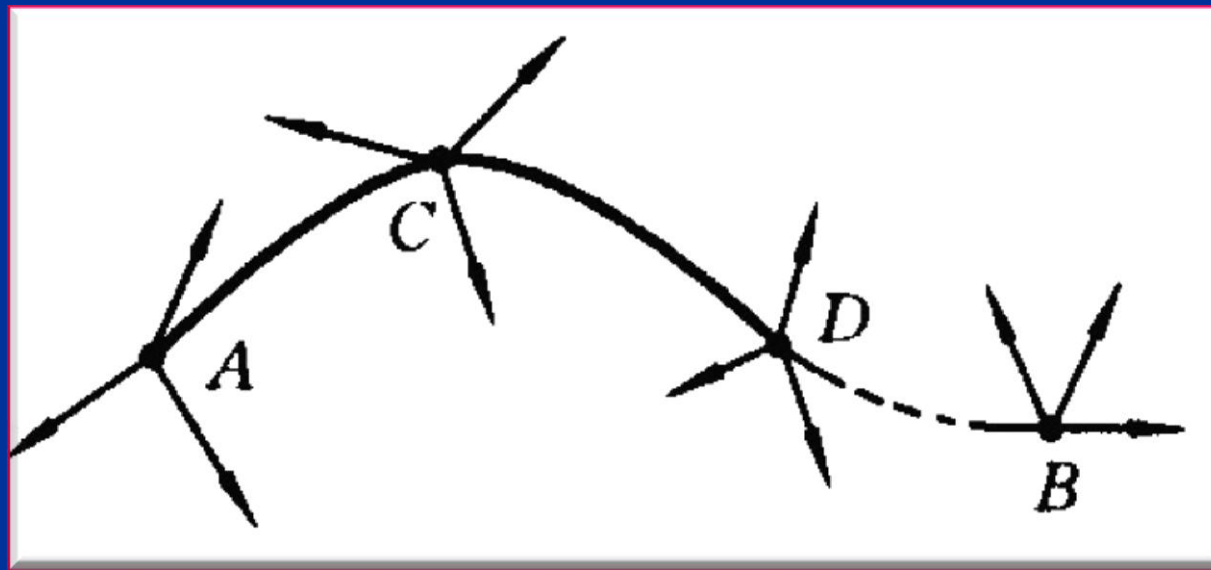


图 7-3 三次多项式插值关节运动轨迹

(a)角位移-时间曲线； (b)角速度-时间曲线； (c)角加速度-时间曲线

2. 过路径点的三次多项式插值

机器人作业除在A、B点有位姿要求外，在路径点C、D…也有位姿要求。对于这种情况，假如终端执行器在路径点停留，即各路径点上速度为0，则轨迹规划可连续直接使用前面介绍的三次多项式插值方法；但如果只是经过，并不停留，就需要将前述方法推广



三次多项式插值

在某段路径上，“起始点”为 θ_0 和 ω_0 ，“终止点”为 θ_f 和 ω_f 。这时，确定三次多项式系数的方法与前面所述的完全一致，只不过是速度约束条件变为

$$\begin{cases} \dot{\theta}(0) = \omega_0 \\ \dot{\theta}(t_f) = \omega_f \end{cases}$$

利用约束条件确定三次多项式系数，有下列方程组：

$$\begin{cases} \theta_0 = a_0 \\ \theta_f = a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 \\ \omega_0 = a_1 \\ \omega_f = a_1 + 2a_2 t_f + 3a_3 t_f^2 \end{cases}$$

三次多项式插值

求解方程组

$$\begin{cases} a_0 = \theta_0 \\ a_1 = \omega_0 \\ a_2 = \frac{3}{t_f^2}(\theta_f - \theta_0) - \frac{1}{t_f}(2\omega_0 + \omega_f) \\ a_3 = -\frac{2}{t_f^3}(\theta_f - \theta_0) + \frac{1}{t_f^2}(\omega_0 + \omega_f) \end{cases}$$

三次多项式插值

路径点上的关节速度，可由以下任一规则确定：

- 如果机器人末端操作器在经过路径点时有速度要求，则可以利用此路径点上的逆雅可比矩阵，把该路径点的直角坐标速度转换成关节坐标速度。轨迹规划时则以此作为速度约束条件。
- 如果某个路径点是机器人的奇异点，即此点的逆雅可比不可求，这时就无法求关节速度了；
- 此外，在求各点关节速度时，要逐点计算逆雅可比矩阵，并依此计算关节速度，耗时较多。

三次多项式插值

路径点上的关节速度，可由以下作

在各路径点上的细实
，其斜率表示过路径
时的关节运动速度。

- 由控制系统采用某种启发式方法自动地选取合适的路径点速度。
- 用三次多项式插值前，先假设各路径点之间关节运动速度是均匀的，即图中所示用直线段将这些路径点依次连接起来。
- 规则选定：如果相邻线段的斜率在路径点处改变符号，则速度选为零；如果相邻线段斜率不改变符号，则选取路径点两侧的线段斜率的平均值作为该点的速度。因此，只要给定路径点，系统就能依此规则自动生成相应的路径点速度。

三次多项式插值

路径点上的关节速度，可由以下任一规则确定：

- 按照保证每个路径点的加速度连续的原则，由控制系统自动地选择路径点的速度。



为此，可以设法用两条三次曲线在路径点处按一定规则连接起来，拼凑成所要求的轨迹。拼凑的约束条件是：连接处速度连续，而且加速度也连续。

第七章 工业机器人轨迹规划与编程

7.1 工业机器人轨迹规划

7.2 关节空间法

7.3 直角坐标空间法（自学）

7.4 轨迹的实时生成（自学）

7.5 工业机器人编程

7.6 工业机器人编程语言（自学）

7.7 工业机器人离线编程

7.5 工业机器人的编程

7.5.1、机器人编程方式

机器人编程，就是针对机器人为完成某项作业进行程序设计

复杂的顺序任务

机器人的作业
任务各不相同

特定环境中作决策

编程能力与编程
方式有很大关系

适应性

作业能力

微型计算机

接近自然语言

接口

人与机器人

实现各种机器人操作

机器人编程方法三种形式：

一个关节对应着示教盒按钮，以分别控制该关节运动。示教盒示教方式适用于大型机器人或危险作业条件示教。

把机器人终端移动至目标位置，并把此机器人关节角度信息记录进内存储器，这

些运动时，顺序控制器从内存读出相应数据，机器人就可重复示教时的轨迹和各种操作，这

手把手示教

示教盒示教

示教盒示教则是利用装在控制盒上的按钮驱动机器人按需要的顺序进行操作。

示教编程优点：

- 只需要简单的设备和控制装置即可进行。
- 操作简单，易于掌握。
- 示教再现过程很快，示教之后马上即可应用。

示教编程缺点：

- 编程占用机器人的作业时间；
- 很难规划复杂的运动轨迹以及准确的直线运动；
- 难以与传感信息相配合；
- 难以与其他操作同步；

机器人编程方法三种形式：

2、机器人语言编程

实现了计算机编程，并可以引入传感信息，从而提供一个更通用的方法来解决人一机器人通信接口问题。目前应用于**工业中的是动作级和对象级机器人语言。**

3、离线编程

用通用语言或专门语言预先进行程序设计，在离线的情况下进行轨迹规划的编程方法。离线编程系统是基于CAD数据的图形编程系统。由于CAD技术的发展，机器人可以利用CAD数据生成机器人路径，这是集机器人于CIMS系统的必由之路。

7.5.2、机器人语言编程

✓早期的工业机器人，由于完成的作业比较简单，作业内容改变不频繁，采用固定程序控制或示教再现方法即可满足要求，不存在语言问题。

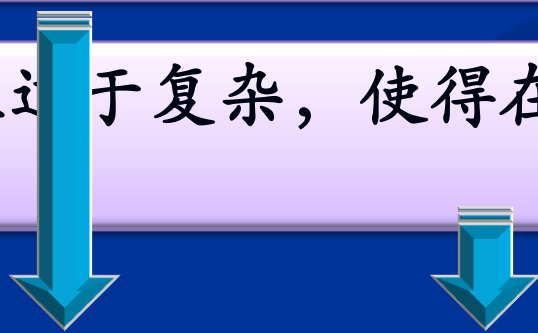
✓机器人本身的发展，计算机系统功能日益完善以及要求机器人作业内容愈加复杂化，利用程序来控制机器人显得越来越困难。

✓编程过程过于复杂，使得在作业现场对付复杂作业十分困难。

描述作业

专用机器人语言

控制机器人动作



(一) 机器人语言的发展

直角坐标机器人

✓ 1973年, 其

扩展BASIC

配作业

PUMA

ME语言

✓ 美国EDM

简单, 易编程

利用传感器位

✓ 19

机床及机

动作描述、力和接触控制

配合视觉系统可完成手眼协调编程

✓ 80, 大

✓ 80^{20#}, 美国麦

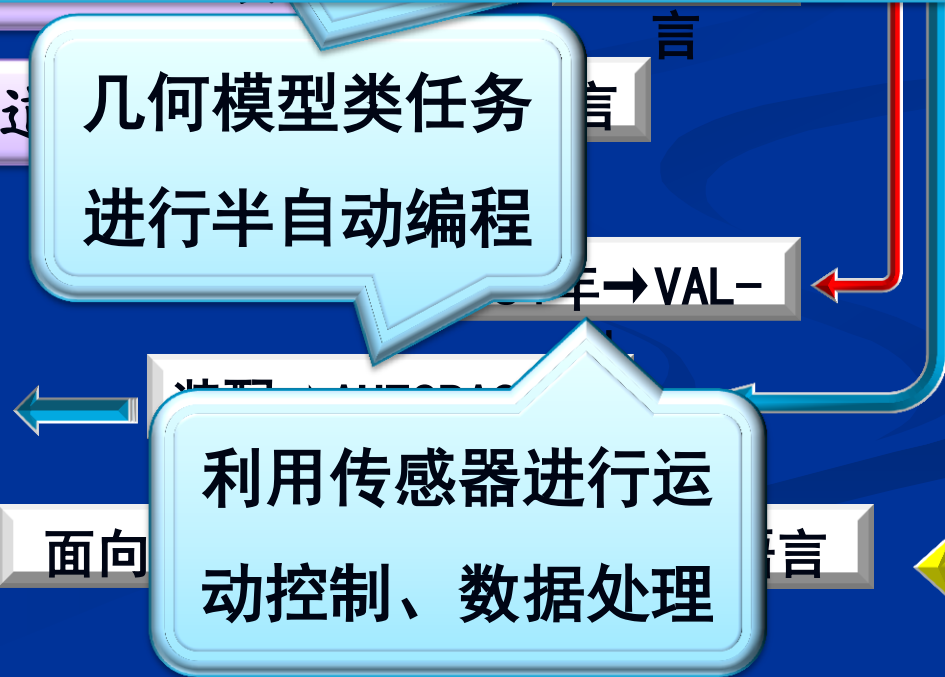
几何模型类任务
进行半自动编程

1982年 → AML语言

面向

利用传感器进行运
动控制、数据处理

语言



(二) 机器人语言的分类

1、根据作业描述水平的高低分

(1) 动作级机器人语言-VAL

以机器人手爪的运动作为作业描述的中心

用该级语言编写的作业程序，通常由使机器人手爪从一个位置到另一个位置的一系列运动语句组成。

动作级机器人语言的每一条语句对应于一个机器人动作。

缺点是不能够进行复杂的运算，仅能接受传感器的开关信号，并且与其他计算机通讯的能力差。

1、根据作业描述水平的高低分

(2) 对象级机器人语言-AUTOPASS

以近似自然语言的方式，按照作业对象的状态变化来进行程序设计。

以描述操作物体之间关系为中心的语言。

它不需要去描述机器人手爪如何动作，只要由操作者给出作业本身的顺序过程的描述及环境模型的描述，机器人即可自行决定如何动作。

有处理传感器信息的能力，可利用传感器信息来修改、更新环境的描述和模型，也可以利用传感器信息进行控制、测试和监督。对象级语言具有良好的开放性，用户可以根据应用的需要扩展语言功能。

1、根据作业描述水平的高低分

(3) 任务级机器人语言

最不

使目
信后

当发出一条“**抓住螺钉**”的指令时，这种语言系统要能进行路径寻找规划，在复杂的环境中找到一条运动路径，沿此路径运动，机器人不会与周围任何障碍物发生碰撞，并能自动进行工件抓取规划，在螺钉上选取一个好的抓取位置，并把螺钉抓起。显然，任务级语言的构成是十分复杂的，它必须具有人工智能的推理系统和大型知识库。

而

作
环
境
，
最
数
据。

须具有判断环境、描述环境的能力；同时，也必须具有自动完成许多规划任务的能力。

2、按表面形式分

汇编型，如VAL语言；

编译型，如AI、LM语言；

自然语言型，如AUTOPASS语言等；

(三) 机器人语言的特征

机器人语言则包含语言本身、语言处理系统和机器人的工作环境模型三部分



机器人**语言**以程序的方式对机器人发出一系列控制命令，**处理系统**则将这些命令进行编译和解释，然后以机器人能接受的指令，控制机器人进行具体操作。

环境模型是对机器人作业环境内的物体位置、姿态以及物体之间关系的表示。

(三) 机器人语言的特征

1. 具有作业环境和作业对象的描述性。

(1) **环境输入**：视机器人语言水平不同，输入方法也不相同。目前的输入方法一般是由操作者与计算机的人机对话来进行的。将来随着视觉技术的发展，可能由机器人视觉的方法自动生成。

(2) **环境建模**：进行机器人编程时，需要描述物体三维空间的几何关系的语言，对操作物体的位置和姿态，操作物体之间的关系进行描述，并使之模型化。

(2) **环境模型的修改、更新**。在作业过程中，操作物体的位置、姿态以及它们之间的关系一般随着作业而发生改变，语言系统要根据操作情况的变化来改变环境模型的内容。

(三) 机器人语言的特征

2. 具有作业内容的描述性

不管作业内容如何复杂，都要能用机器人语言准确进行描述。

可以用语句命令，也可以用自然语言形式来描述。作业的描述与环境有密切联系，而且描述的水平决定了机器人语言水平。

目前的机器人语言一般需要给出作业顺序，使用的方法是通过利用语法和语义定义完善的输入语言，由它给出作业步骤。

(三) 机器人语言的特征

3. 具有良好的编程环境

与通用计算机高级语言一样，良好的编程环境有助于提高编程效率。

编程基础软件，如文本编辑、调试程序和文件系统等都是必不可少的。

由于对机器人编程，调试工作相当麻烦，除了要保证机器人运动顺序正确，还要保证运动的精度，反复调试不可避免，所以总希望采用方便的交互式编程方法。

(三) 机器人语言的特征

4. 具有人机接口和传感器接口功能

在编制程序和作业过程中，要便于人与机器人之间进行信息交换，以及在运动出现故障时能及时处理。

随着作业环境和作业内容复杂程度的增加，需要有功能强大的人机接口。利用传感器接口，对传感器的信息进行综合，提供决策和控制命令。

第七章 工业机器人轨迹规划与编程

7.1 工业机器人轨迹规划

7.2 关节空间法

7.3 直角坐标空间法（自学）

7.4 轨迹的实时生成（自学）

7.5 工业机器人编程

7.6 工业机器人编程语言（自学）

7.7 工业机器人离线编程

7.7 机器人离线编程

7.7.1 离线编程的概念

第一代工业机器人采用示教编程方式，无论是采用手把手示教或控制盒示教，都需要机器人停止原来的工作。而再现时若不能满足要求，还需反复进行示教。



进行一项任务之前，在现场编程过程要花费很多时间，这对于大批量生产的简单作业，基本还能满足要求。但是，随着机器人应用到中小批量生产，以及要求完成任务的复杂程度的增加，用示教编程方式就很难适应了。

7.7.1 离线编程的概念

随着计算机技术和机器人技术的不断发展，机器人与CAD/CAM技术结合，已形成生产效率很高的柔性制造系统(FMS)和计算机集成制造系统(CIMS)。这些系统中大量采用工业机器人，具有很高的适用性和灵活性。



在这样的环境中，若仍采用示教编程方式，当对某台机器人进行编程或修改程序时，就得让上整个生产线都停顿下来。

在复杂环境中工业机器人，在实际使用之前，对机器人及其工作过程的计算机仿真是必不可少的。

7.7.1 离线编程的概念

离线编程与机器人语言编程。



语言编程目前是动作级机器人语言和对象级机器人语言，编程工作非常繁重。

机器人离线编程就是利用计算机图形学的成果，建立机器人及作业环境的三维几何模型，然后对机器人所要完成的任务进行离线规划和编程，并对编程结果进行动态图形仿真，最后将满足要求的编程结果传到机器人控制柜，使机器人完成指定的作业任务。

离线编程可以看作动作级和对象级语言图形方式的延伸，是研制任务级语言编程的重要基础。

7.7.2 离线编程系统的一般要求

工业机器人离线编程系统的一个重要特点是能够与CAD/CAM建立联系，能够利用CAD数据库的资料。对于一个简单的机器人作业，几乎可以直接利用CAD对零件的描述来实现编程。

7.7.2 离线编程系统的一般要求

- 对将要编程的生产系统工作过程的全面了解；
- 机器人和工作环境三维实体模型；
- 机器人几何学、运动学和动力学的知识；
- 能用专门语言或通用语言编写出基于(1)、(2)、(3)的软件系统，要求该系统是基于图形显示的；
- 能用计算机构型系统进行动态模拟仿真，对运动程序进行测试，并检测算法，如检查机器人关节角超限，运动轨迹是否正确，以及进行碰撞的检测；

7.7.2 离线编程系统的一般要求

- 传感器的接口和仿真，以利用传感器的信息进行决策和规划；

- 通信功能，从离线编程系统所生成的运动代码到各种机器人控制柜的通信；

- 用户接口，提供友好的人/机界面，并要解决好计算机与人机接口问题，以便人工干预和进行系统的

校

正！

离线编程系统通过仿真模拟机器人系统的图形模型，通过实际环境中的运动而进行编程的，存在模型与实际情况的**误差**。

7.7.3 离线编程系统的基本组成

用户接口又称用户界面，是计算机与用户之间通
机器人系统的三维几何构型几何描述和图形显示，
机器人的**运动学**计算包含两部分：一是运动学正解，
轨迹规划是用来生成关节空间或直角空间的轨迹，
机器人动态仿真它逼真地模拟了机器人的实际工作
过程，为程序员提供了直观的可视图形。进而可以
协议之间的相互认同，它能把机器人仿真程序直接
转化成各种机器人控制器能接受的代码。后置加工
是指对语言加工或翻译，使离线编程系统结果转换
成机器人控制器可接受的格式或代码。

仿真模型和被仿真的实际机器人之间存在误差，故
在离线编程系统中要设置**误差校正环节**。如何有效
地消除或减小误差，是离线编程系统实用化的关键。