

## 第3章 计算机数控装置的插补原理

### 3.1 概 述

#### 3.1.1 插补的基本概念

在数控机床中，刀具或工件的最小位移量是机床坐标轴运动的一个分辨单位，由检测装置辨识，称为分辨率(闭环系统)，或称为脉冲当量(开环系统)，又叫做最小设定单位。因此刀具的运动轨迹在微观上是由小线段构成的折线，不可能绝对地沿着刀具所要求的零件廓形运动，只能用折线逼近所要求的廓形曲线。机床数控系统依据一定方法确定刀具运动轨迹，进而产生基本廓形曲线，如直线、圆弧等。其它需要加工的复杂曲线由基本廓形曲线逼近，这种拟合方法称为“插补”(Interpolation)。“插补”实质是数控系统根据零件轮廓线型的有限信息(如直线的起点、终点,圆弧的起点、终点和圆心等)，计算出刀具的一系列加工点、完成所谓的数据“密化”工作。插补有两层意思：一是产生基本线型，二是用基本线型拟合其它轮廓曲线。插补运算具有实时性，要满足刀具运动实时控制的要求。其运算速度和精度会直接影响数控系统的性能指标。

#### 3.1.2 插补方法的分类

数控系统中完成插补运算的装置或程序称为插补器，根据插补器的结构可分为硬件插补器、软件插补器和软、硬件结合插补器三种类型。早期 NC 系统的插补运算由硬接线的数字电路装置来完成，称为硬件插补，其结构复杂，成本较高。在 CNC 系统中插补功能一般由计算机程序来完成，称为软件插补。由于硬件插补具有速度高的特点，为了满足插补速度和精度的要求，现代 CNC 系统也采用软件与硬件相结合的方法，由软件完成粗插补，由硬件完成精插补。

由于直线和圆弧是构成零件轮廓的基本线型，因此 CNC 系统一般都具有直线插补和圆弧插补两种基本类型，在三坐标以上联动的 CNC 系统中，一般还具有螺旋线插补和其它线型插补。为了方便对各种曲线、曲面的直接加工，人们一直研究各种曲线的插补功能，在一些高档 CNC 系统中，已经出现了抛物线插补、渐开线插补、正弦线插补、样条曲线插补、球面螺旋线插补以及曲面直接插补等功能。

插补运算所采用的原理和方法很多，一般可归纳为基准脉冲插补和数据采样插补两大类。

##### 1. 基准脉冲插补

基准脉冲插补又称为脉冲增量插补或行程标量插补，其特点是每次插补结束仅向各运动坐标轴输出一个控制脉冲，因此各坐标仅产生一个脉冲当量或行程的增量。脉冲序列的频率代表坐标运动的速度，而脉冲的数量代表运动位移的大小。这类插补运算简单，容易用硬件电路来实现，早期的硬件插补都是采用这类方法，在目前的 CNC 系统中原来的硬件插补功能可以用软件来实现，但仅适用于一些中等速度和中等精度的系统，主要用于步进电机驱动的开环系统。也有的数控系统将其用做数据采样插补中的精插补。

基准脉冲插补的方法很多，如逐点比较法、数字积分法、脉冲乘法器、矢量判别法、比较积分法、最小偏差法、单步追踪法等等。应用较多的是逐点比较法和数字积分法。

## 2. 数据采样插补

数据采样插补又称数字增量插补、时间分割插补或时间标量插补，其运算采用时间分割思想，根据编程的进给速度将轮廓曲线分割为每个插补周期的进给直线段（又称轮廓步长），以此来逼近轮廓曲线。数控装置将轮廓步长分解为各坐标轴的插补周期进给量，作为命令发送给伺服驱动装置。伺服系统按位移检测采样周期采集实际位移量，并反馈给插补器进行比较完成闭环控制。伺服系统中指令执行过程实质也是数据密化工作。闭环或半闭环控制系统都采用数据采样插补方法，它能满足控制速度和精度的要求。数据采样插补方法很多，如直线函数法、扩展数字积分法、二阶递归算法等。但都基于时间分割的思想。

# 3.2 基准脉冲插补

## 3.2.1 逐点比较法

### 1. 插补原理及特点

逐点比较法是我国早期数控机床中广泛采用的一种方法，又称代数运算法或醉步法。其基本原理是每次仅向一个坐标轴输出一个进给脉冲，而每走一步都要通过偏差函数计算，判断偏差点的瞬时坐标同规定加工轨迹之间的偏差，然后决定下一步的进给方向。每个插补循环由偏差判别、进给、偏差函数计算和终点判别四个步骤组成。逐点比较法可进行直线插补、圆弧插补，也可用于其它曲线的插补，其特点是运算直观，插补误差不大于一个脉冲当量，脉冲输出均匀，调节方便。

### 2. 逐点比较法直线插补

#### (1) 偏差函数构造

直线插补时，通常将坐标原点设在直线起点上。对于第一象限直线  $OA$ ，如图 3.1 所示，其方程可表示为：

$$\frac{X}{Y} - \frac{X_e}{Y_e} = 0$$

改写为

$$YX_e - XY_e = 0$$

若刀具加工点为  $P_i(X_i, Y_i)$ ，则该点的偏差函数  $F_i$  可表示为：

$$F_i = Y_i X_e - X_i Y_e \quad (3.1)$$

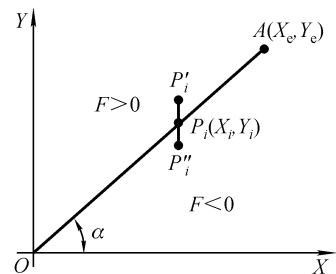


图 3.1 逐点比较法直线插补

若  $F_i = 0$ ，表示加工点位于直线上；  
 若  $F_i > 0$ ，表示加工点位于直线上方；  
 若  $F_i < 0$ ，表示加工点位于直线下方。

(2) 偏差函数数字的递推计算

为了简化式(3.1)的计算，通常采用偏差函数的递推式(迭代式)。

若  $F_i > 0$ ，规定向  $+X$  方向走一步，若坐标单位用脉冲当量表示，则有

$$\begin{cases} X_{i+1} = X_i + 1 \\ F_{i+1} = X_e Y_i - Y_e (X_i + 1) = F_i - Y_e \end{cases} \quad (3.2)$$

若  $F_i < 0$ ，规定  $+Y$  方向走一步，则有

$$\begin{cases} Y_{i+1} = Y_i + 1 \\ F_{i+1} = X_e (Y_i + 1) - Y_e X_i = F_i + X_e \end{cases} \quad (3.3)$$

因此插补过程中用式(3.2)和式(3.3)代替式(3.1)进行偏差计算，可使计算大为简化。

(3) 终点判别

直线插补的终点判别可采用三种方法：

- 1) 判断插补或进给的总步数： $N = X_e + Y_e$ ；
- 2) 分别判断各坐标轴的进给步数；
- 3) 仅判断进给步数较多的坐标轴的进给步数。

(4) 逐点比较法直线插补举例

对于第一象限直线  $OA$ ，终点坐标  $X_e = 6$ ， $Y_e = 4$ ，其插补运算过程如表 3.1 所示，插补轨迹如图 3.2 所示。插补从直线起点  $O$  开始，故  $F_0 = 0$ 。终点判别是判断进给总步数  $N = 6 + 4 = 10$ ，将其存入终点判别计数器中，每进给一步减 1，若  $N = 0$ ，则停止插补。

表 3.1 逐点比较法直线插补过程

步 数	偏 差 判 别	坐 标 进 给	偏 差 计 算	终 点 判 别
0			$F_0 = 0$	$\Sigma = 10$
1	$F = 0$	$+X$	$F_1 = F_0 - y_e = 0 - 4 = -4$	$\Sigma = 10 - 1 = 9$
2	$F < 0$	$+Y$	$F_2 = F_1 + x_e = -4 + 6 = 2$	$\Sigma = 9 - 1 = 8$
3	$F > 0$	$+X$	$F_3 = F_2 - y_e = 2 - 4 = -2$	$\Sigma = 8 - 1 = 7$
4	$F < 0$	$+Y$	$F_4 = F_3 + x_e = -2 + 6 = 4$	$\Sigma = 7 - 1 = 6$
5	$F > 0$	$+X$	$F_5 = F_4 - y_e = 4 - 4 = 0$	$\Sigma = 6 - 1 = 5$
6	$F = 0$	$+X$	$F_6 = F_5 - y_e = 0 - 4 = -4$	$\Sigma = 5 - 1 = 4$
7	$F < 0$	$+Y$	$F_7 = F_6 + x_e = -4 + 6 = 2$	$\Sigma = 4 - 1 = 3$
8	$F > 0$	$+X$	$F_8 = F_7 - y_e = 2 - 4 = -2$	$\Sigma = 3 - 1 = 2$
9	$F < 0$	$+Y$	$F_9 = F_8 + x_e = -2 + 6 = 4$	$\Sigma = 2 - 1 = 1$
10	$F > 0$	$+X$	$F_{10} = F_9 - y_e = 4 - 4 = 0$	$\Sigma = 1 - 1 = 0$

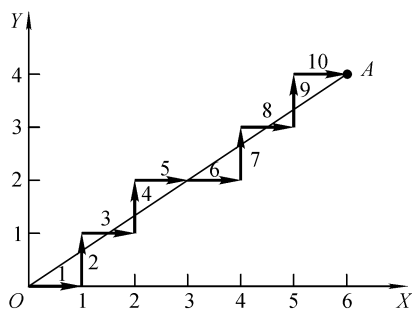


图 3.2 逐点比较法直线插补轨迹

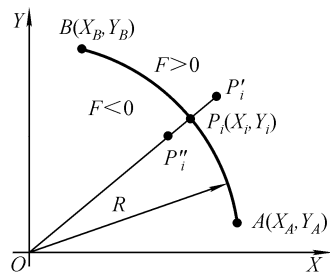


图 3.3 逐点比较法圆弧插补

### 3. 逐点比较法圆弧插补

#### (1) 偏差函数构造

若加工半径为  $R$  的圆弧  $AB$ ，将坐标原点定在圆心上，如图 3.3 所示。对于任意加工点  $P_i(X_i, Y_i)$ ，其偏差函数  $F_i$  可表示为：

$$F_i = X_i^2 + Y_i^2 - R^2 \quad (3.4)$$

显然，若  $F_i = 0$ ，表示加工点位于圆上；若  $F_i > 0$ ，表示加工点位于圆外；若  $F_i < 0$ ，表示加工点位于圆内。

#### (2) 偏差函数的递推计算

为了简化式(3.4)的计算，需采用其递推式(或迭代式)。圆弧加工可分为顺时针加工或逆时针加工，与此相对应的便有逆圆插补和顺圆插补两种方式，下面就第一象限圆弧，对其递推式公式加以推导。

##### 1) 逆圆插补

若  $F \geq 0$ ，规定向  $-X$  方向走一步，

$$\begin{cases} X_{i+1} = X_i - 1 \\ F_{i+1} = (X_i - 1)^2 + Y_i^2 - R^2 = F_i - 2X_i + 1 \end{cases} \quad (3.5)$$

若  $F_i < 0$ ，规定向  $+Y$  方向走一步，则有

$$\begin{cases} Y_{i+1} = Y_i + 1 \\ F_{i+1} = X_i^2 + (Y_i + 1)^2 - R^2 = F_i + 2Y_i + 1 \end{cases} \quad (3.6)$$

##### 2) 顺圆插补

若  $F_i \geq 0$ ，规定向  $-Y$  方向走一步，则有

$$\begin{cases} Y_{i+1} = Y_i - 1 \\ F_{i+1} = X_i^2 + (Y_i - 1)^2 - R^2 = F_i - 2Y_i + 1 \end{cases} \quad (3.7)$$

若  $F_i < 0$ ，规定向  $+X$  方向走一步，则有

$$\begin{cases} X_{i+1} = X_i + 1 \\ F_{i+1} = (X_i + 1)^2 + Y_i^2 - R^2 = F_i + 2X_i + 1 \end{cases} \quad (3.8)$$

#### (3) 终点判别

终点判别可采用与直线插补相同的方法：

- 1) 判断插补或进给的总步数:  $N = |X_a - X_b| + |Y_a - Y_b|$  ;
- 2) 分别判断各坐标轴的进给步数:  $N_x = |X_a - X_b|$  ,  $N_y = |Y_a - Y_b|$  。

(4) 逐点比较法圆弧插补举例

对于第一象限圆弧  $AB$ , 起点  $A(4,0)$ , 终点  $B(0,4)$ , 如图 3.4 所示。要求采用逆圆插补方法, 其运算过程如表 3.2 所示, 插补轨迹如图 3.4 所示。

表 3.2 逐点比较法圆弧插补过程

步数	偏差判别	坐标进给	偏差计算	坐标计算	终点判别
起点			$F_0 = 0$	$x_0 = 4, y_0 = 0$	$\Sigma = 4 + 4 = 8$
1	$F_0 < 0$	$-x$	$F_1 = F_0 - 2x_0 + 1$ $= 0 - 2 * 4 + 1 = -7$	$x_1 = 4 - 1 = 3$ $y_1 = 0$	$\Sigma = 8 - 1 = 7$
2	$F_1 < 0$	$+y$	$F_2 = F_1 + 2y_1 + 1$ $= -7 + 2 * 0 + 1 = -6$	$x_2 = 3$ $y_2 = y_1 + 1 = 1$	$\Sigma = 7 - 1 = 6$
3	$F_2 < 0$	$+y$	$F_3 = F_2 + 2y_2 + 1 = -3$	$x_3 = 3, y_3 = 2$	$\Sigma = 5$
4	$F_3 < 0$	$+y$	$F_4 = F_3 + 2y_3 + 1 = 2$	$x_4 = 3, y_4 = 3$	$\Sigma = 4$
5	$F_4 > 0$	$-x$	$F_5 = F_4 - 2x_4 + 1 = -3$	$x_5 = 2, y_5 = 3$	$\Sigma = 3$
6	$F_5 < 0$	$+y$	$F_6 = F_5 + 2y_5 + 1 = 4$	$x_6 = 2, y_6 = 4$	$\Sigma = 2$
7	$F_6 > 0$	$-x$	$F_7 = F_6 - 2x_6 + 1 = 1$	$x_7 = 1, y_7 = 4$	$\Sigma = 1$
8	$F_7 < 0$	$-x$	$F_8 = F_7 - 2x_7 + 1 = 0$	$x_8 = 0, y_8 = 4$	$\Sigma = 0$

4. 逐点比较法的速度分析

刀具进给速度是插补方法的重要性能指标, 也是选择插补方法的重要依据。

(1) 直线插补的速度分析

直线加工时, 有

$$\frac{L}{v} = \frac{N}{f}$$

式中:  $L$ —直线长度;  $v$ —刀具进给速度;  $N$ —插补循环数;  $f$ —插补脉冲的频率。

$$N = X_e + Y_e = L \cos \alpha + L \sin \alpha$$

式中:  $\alpha$ —直线与  $X$  轴的夹角。

则

$$v = \frac{f}{\sin \alpha + \cos \alpha} \tag{3.9}$$

式(3.9)说明刀具进给速度与插补时钟频率  $f$  和与  $X$  轴夹角  $\alpha$  有关。若保持  $f$  不变, 加工  $0^\circ$  和  $90^\circ$  倾角的直线时刀具进给速度最大(为  $f$ ), 加工  $45^\circ$  倾角直线时速度最小(为  $0.707f$ ), 如图 3.5 所示。

(2) 圆弧插补的速度分析

如图 3.6 所示,  $P$  是圆弧  $AB$  上任意一点,  $cd$  是圆弧在  $P$  点的切线, 切线与  $X$  轴夹角为  $\alpha$ 。

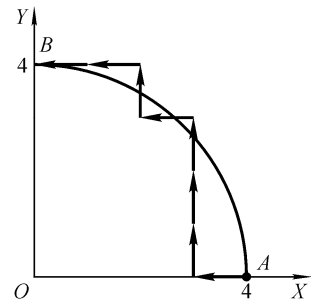


图 3.4 逐点比较法圆弧插补轨

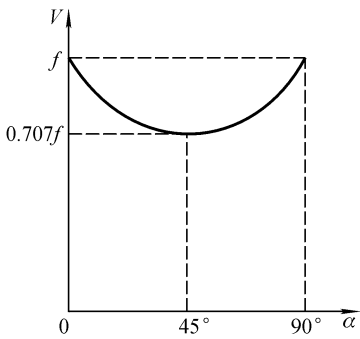


图 3.5 逐点比较法直线插补速度的变化

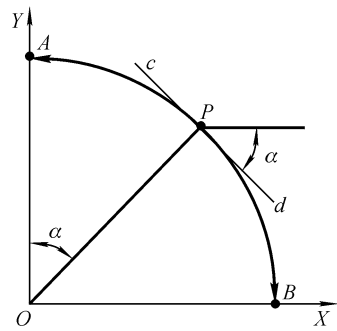


图 3.6 逐点比较法圆弧插补速度分析

显然刀具在  $P$  点的速度可认为与插补切线  $cd$  的速度基本相等，因此，由式(3.9)可知加工圆弧时刀具的进给速度是变化的，除了与插补时钟的频率成正比外，还与切削点处的半径同  $Y$  轴的夹角  $\alpha$  有关，在  $0^\circ$  和  $90^\circ$  附近进给速度最快(为  $f$ )，在  $45^\circ$  附近速度为最慢(为  $0.707f$ )，进给速度在  $(1 \sim 0.707)f$  间变化。

### 5. 逐点比较法的象限处理

以上仅讨论了第一象限的直线和圆弧插补，对于其它象限的直线和圆弧，可采取不同方法进行处理。下面介绍其中的两种。

#### (1) 分别处理法

前面讨论的插补原理与计算公式，仅适用与第一象限的情况。对于其它象限的直线插补和圆弧插补，可根据上面的分析方法，分别建立其偏差函数的计算公式。这样对于四个象限的直线插补，会有 4 组计算公式，对于 4 个象限的逆时针圆弧插补和 4 个象限的顺时针圆弧插补，会有 8 组计算公式，其刀具的偏差和进给方向可用图 3.7 的简图加以表示。

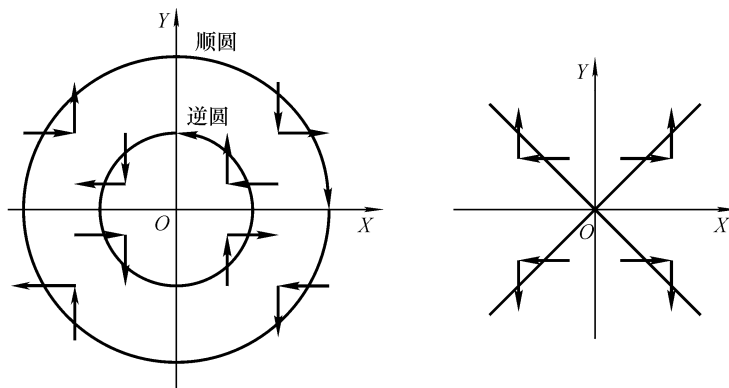


图 3.7 圆弧与直线插补中刀具偏差与进给方向简图

#### (2) 坐标变换法

通过对图 3.6 进行分析，可以发现能通过坐标变换减少偏差函数计算公式。若将原坐标系  $OXY$  变换为  $O'X'Y'$ ，且

$$\begin{cases} X = |X'| \\ Y = |Y'| \end{cases} \quad (3.10)$$

则可用第一象限的直线插补的偏差函数完成其余三个象限直线插补的偏差计算, 用第一象限逆圆插补的偏差函数进行第三象限逆圆和第二、四象限顺圆插补的偏差计算, 用第一象限顺圆插补的偏差函数进行第三象限顺圆和第二、四象限逆圆插补的偏差计算。

### 3.2.2 数字积分法

#### 1. 插补原理及其特点

数字积分法又称 DDA (Digital Differential Analyzer) 法, 是利用数字积分的方法, 计算刀具沿各坐标轴的位移, 以便加工出所需要的线型。

若加工图 3.10(在后面)的圆弧  $AB$ , 刀具在  $X$ 、 $Y$  轴方向的速度必须满足:

$$\begin{cases} v_X = v \cos \alpha \\ v_Y = v \sin \alpha \end{cases}$$

式中:  $v_X$ 、 $v_Y$ —刀具在  $X$ 、 $Y$  轴方向的进给速度;

$v$ —刀具沿圆弧运动的切线速度;

$\alpha$ —圆弧上任一点处切线同  $X$  轴的夹角。

用积分法可以求得刀具在  $X$ 、 $Y$  方向的位移, 即

$$X = \int v_X dt = \int v \cos \alpha dt$$

$$Y = \int v_Y dt = \int v \sin \alpha dt$$

其数字积分表达式为

$$\begin{cases} X = \sum v_X \Delta t = \sum v \cos \alpha \Delta t \\ Y = \sum v_Y \Delta t = \sum v \sin \alpha \Delta t \end{cases} \quad (3.11)$$

式中:  $\Delta t$ —插补循环周期。

可见只要能求出曲线的切线方向, 便可对曲线进行插补。数字积分法运算速度快、脉冲分配均匀, 易于实现各种曲线、特别是多坐标空间曲线的插补, 因此应用广泛。

#### 2. DDA 法直线插补

##### (1) DDA 法直线插补的积分表达式

对于图 3.8 所示的直线  $OA$ , 有

$$\frac{v}{L} = \frac{v_X}{X_e} = \frac{v_Y}{Y_e} = K \quad (3.12)$$

式中:  $L$ —直线长度;  $K$ —比例系数。

则  $v_X = KX_e$ ,  $v_Y = KY_e$ , 代入式(3.11)得

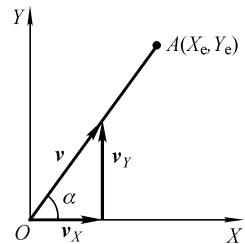


图 3.8 DDA 法直线插补原理

$$\begin{cases} X = K \sum_{i=1}^m X_e \Delta t \\ Y = K \sum_{i=1}^m Y_e \Delta t \end{cases} \quad (3.13)$$

令  $\Delta t = 1$ ,  $K = \frac{1}{2^N}$ , 则

$$\begin{cases} X = \sum_{i=1}^m \frac{X_e}{2^N} \\ Y = \sum_{i=1}^m \frac{Y_e}{2^N} \end{cases} \quad (3.14)$$

式中:  $N$ —积分累加器的位数。

式(3.14)便是 DDA 直线插补的积分表达式。因为  $N$  位累加器的最大存数为  $2^N - 1$ , 当累加数等于或大于  $2^N$  时, 便发生溢出, 而余数仍存放在累加器中。这种关系式还可以表示为:

$$\text{积分值} = \text{溢出脉冲数代表的值} + \text{余数}$$

当两个积分累加器根据插补时钟进行同步累加时, 溢出脉冲数必然符合式(3.13), 用这些溢出脉冲数分别控制相应坐标轴的运动, 必然能加工出所要求的直线。 $X_e$ 、 $Y_e$  又称作积分函数, 而积分累加器又称为余数寄存器。DDA 法直线插补的进给方向的判别比较简单, 因为插补是从直线起点即坐标原点开始, 坐标轴的进给方向总是直线终点坐标绝对值增加的方向。

### (2) 终点判别

若累加次数  $m = 2^N$ , 由式(3.14)可得:

$$X = \frac{1}{2^N} \sum_{i=1}^{2^N} X_e = X_e, \quad Y = \frac{1}{2^N} \sum_{i=1}^{2^N} Y_e = Y_e \quad (3.15)$$

因此, 累加次数、即插补循环数是否等于  $2^N$  可作为 DDA 法直线插补终点判别的依据。

### (3) DDA 法直线插补举例

插补第一象限直线  $OA$ , 起点为  $O(0,0)$ , 终点为  $A(5,3)$ 。取被积函数寄存器分别为  $J_{VX}$ 、 $J_{VY}$ , 余数寄存器分别为  $J_{RX}$ 、 $J_{RY}$ , 终点计数器为  $J_E$ , 均为三位二进制寄存器。插补过程如表 3.3 所示, 插补轨迹如图 3.9 所示。从图中可以看出, DDA 法允许向两个坐标轴同时发出进给脉冲, 这一点与逐点比较法不同。

表 3.3 DDA 直线插补过程

累加次数 ( $\Delta t$ )	X 积分器			Y 积分器			终点计数器 $J_E$	备注
	$J_{VX}$ ( $X_e$ )	$J_{RX}$	溢出 $\Delta X$	$J_{VY}$ ( $Y_e$ )	$J_{RY}$	溢出 $\Delta Y$		
0	101	000		011	000		000	初始状态
1	101	101		011	011		001	第一次迭代
2	101	010	1	011	110		010	$J_{RX}$ 有进位, $\Delta X$ 溢出脉冲
3	101	111		011	001	1	011	$J_{RY}$ 有进位, $\Delta Y$ 溢出脉冲

续表

累加次数 ( $\Delta t$ )	X 积分器			Y 积分器			终点计数器 $J_E$	备注
	$J_{VX}$ ( $X_e$ )	$J_{RX}$	溢出 $\Delta X$	$J_{VY}$ ( $Y_e$ )	$J_{RY}$	溢出 $\Delta Y$		
4	101	100	1	011	100		100	$\Delta X$ 溢出
5	101	001	1	011	111		101	$\Delta X$ 溢出
6	101	110		011	010	1	110	$\Delta Y$ 溢出
7	101	011	1	011	101		111	$\Delta X$ 溢出
8	101	000	1	011	000	1	000	$\Delta X$ 、 $\Delta Y$ 同时溢出 $J_E = 0$ , 插补结束

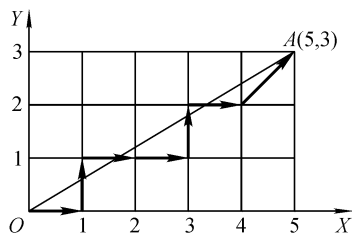


图 3.9 DDA 法直线插补轨迹

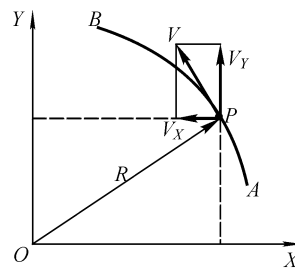


图 3.10 DDA 圆弧插补原理

### 3. DDA 法圆弧插补

#### (1) DDA 法圆弧插补的积分表达式

对图 3.10 所示的第一象限圆弧，圆心  $O$  位于坐标原点，半径为  $R$ ，两端点为  $A(X_A, Y_A)$ 、 $B(X_B, Y_B)$ ，刀具位置为  $P(X_i, Y_i)$ ，若采用逆时针加工，有

$$\frac{v}{R} = \frac{v_x}{Y_i} = \frac{v_y}{X_i} = K \quad (3.16)$$

则  $v_x = KY_i$ ， $v_y = KX_i$ ，代入式(3.11)中，

令  $\Delta t = 1$ ， $K = \frac{1}{2^N}$  ( $N$  为累加器的位数)，则有：

$$\begin{cases} X = \frac{1}{2^N} \sum_{i=1}^m Y_i \\ Y = \frac{1}{2^N} \sum_{i=1}^m X_i \end{cases} \quad (3.17)$$

显然用 DDA 法进行圆弧插补时，是对切削点的即时坐标  $X_i$  与  $Y_i$  的数值分别进行累加，若累加器产生溢出，则在相应坐标方向进给一步，进给方向则必须根据刀具的切向运动方向在坐标轴上的投影方向来确定，即决定于圆弧所在象限和顺圆或逆圆插补，如表 3.4 所示。

表 3.4 DDA 法圆弧插补的进给方向

插补方向	顺 圆				逆 圆			
	I	II	III	IV	I	II	III	IV
$\Delta X$	+	+	-	-	-	-	+	+
$\Delta Y$	-	+	+	-	+	-	-	+

(2) 终点判别

DDA 法圆弧插补的终点判别不能通过插补运算的次数来判别，而必须根据进给次数来判别。由于利用两坐标方向进给的总步数进行终点判别时，会引起圆弧终点坐标出现大于一个脉冲当量，但小于两个脉冲当量的偏差，偏差较大，一般采用分别判断各坐标方向进给步数的方法，即  $N_X = |X_A - X_B|$ ， $N_Y = |Y_A - Y_B|$ 。

(3) DDA 圆弧插补举例

对于第 1 象限圆弧，两端点为  $A(5,0)$  和  $B(0,5)$ ，采用逆时针圆弧插补，插补脉冲计算过程如表 3.5 所示，插补轨迹如图 3.11 所示。因为插补中要对刀具位置坐标数值进行累加，因此一旦累加器发生溢出，即说明刀具在相应坐标方向走了一步，则必须对其坐标值、即被积函数进行修改。该例中，两坐标的进给步数均为 5。在插补中，一旦某坐标进给步数达到了要求，则停止该坐标方向的插补运算。

表 3.5 DDA 圆弧插补过程

运算次序	X 积分器			X 终	Y 积分器			Y 终	备 注
	$J_{VX}$ ( $Y_i$ )	$J_{RX}$ ( $\sum Y_i$ )	$\Delta X$		$J_{VY}$ ( $X_i$ )	$J_{RY}$ ( $\sum X_i$ )	$\Delta Y$		
0	000	000	0	101	101	000	0	101	初始状态
1	000	000	0	101	101	101	0	101	第一次迭代
2	000	000	0	101	101	010	1	100	产生 $\Delta Y$ 修正 $Y_i$
	001								
3	001	001	0	101	101	111	0	100	
4	001	010	0	101	101	100	1	011	积分器再次溢出 修正 $J_{VX}$ (即修 $Y_i$ )
	010								
5	010	100	0	101	101	001	1	010	产生 $\Delta Y$ 修正 $Y_i$
	011								
6	011	111	0	101	101	110	0	010	
7	011	010	1	100	101	011	1	001	同时产生 $\Delta X$ 、 $\Delta Y$ 修正 $X_i$ 、 $Y_i$
	100				100				
8	100	110	0	100	100	111	0	001	

续表

运算次序	X 积分器			X 终	Y 积分器			Y 终	备注
	$J_{VX}$ ( $Y_i$ )	$J_{RX}$ ( $\sum Y_i$ )	$\Delta X$		$J_{VY}$ ( $X_i$ )	$J_{RY}$ ( $\sum X_i$ )	$\Delta Y$		
9	100	010	1	011	100	011	1	000	产生 $\Delta X$ 、 $\Delta Y$ ， Y 到达终点，停止 Y 迭代
	101				011				
10	101	111	0	011	011				
11	101	001	1	001	011				产生 $\Delta X$ 修正 $X_i$ 值
					010				
12	101	001	1	001	010				产生 $\Delta X$ 修正 $X_i$
					001				
13	101	110	0	001	001				
14	101	011	1	000	001				产生 $\Delta X$ ，X 终点 到插补结束
					000				

#### 4. DDA 法插补的速度分析

由式(3.12)和式(3.16)，可将直线插补与圆弧插补时的进给速度分别表示为

$$\begin{aligned} v &= \frac{1}{2^N} L f \delta \\ v &= \frac{1}{2^N} R f \delta \end{aligned} \quad (3.18)$$

式中： $f$ —插补时钟频率；

$\delta$ —坐标轴的脉冲当量。

显然，进给速度受到被加工直线的长度和被加工圆弧的半径的影响，特别是行程长且走刀快，行程短且走刀慢，引起各程序段进给速度的不一致，影响加工质量和加工效率，为此人们采取了许多改善措施。

##### (1) 设置进给速率数 FRN

利用 G93，设置进给速率数 FRN(Feed Rate Number)，即

$$\begin{cases} \text{FRN} = \frac{v}{L} = \frac{1}{2^N} f \delta \\ \text{FRN} = \frac{v}{R} = \frac{1}{2^N} f \delta \end{cases} \quad (3.19)$$

则  $v = \text{FRN} \cdot L$ ，或  $v = \text{FRN} \cdot R$ ，通过 FRN 调整插补时钟频率  $f$ ，使其与给定的进给速度相协调，消除线长  $L$  与圆弧半径  $R$  对进给速度的影响。

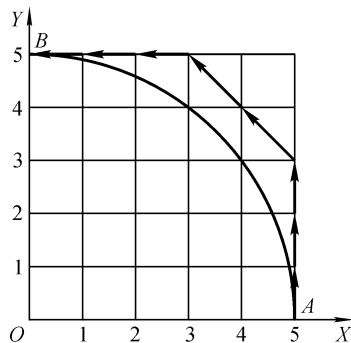


图 3.11 DDA 圆弧插补轨迹

## (2) 左移规格化

所谓左移规格化是当被积函数过小时将被积函数寄存器中的数值同时左移，使两个方向的脉冲分配速度扩大同样的倍数而两者的比值不变，提高加工效率，同时还会使进给脉冲变得比较均匀。

直线插补时，左移的位数要使坐标值较大的被积函数寄存器的最高有效位数为 1，保证每经过两次累加运算必有一次溢出。

圆弧插补时，左移的位数要使坐标值较大的被积函数寄存器的次高位为 1，以保证被积函数修改时不致直接导致溢出。

# 3.3 数据采样插补

## 3.3.1 概述

### 1. 数据采样插补的基本原理

对于闭环和半闭环控制的系统，其分辨率较小，( $\leq 0.001$  mm)，运行速度较高，加工速度高达 24 m/min，以至更高。若采用基本脉冲插补，计算机要执行 20 多条指令，约需 40  $\mu$ s 的时间，而产生的仅是一个控制脉冲，坐标轴仅移动一个脉冲当量，这样一来计算机根本无法执行其它任务，因此必须采用数据采样插补。

数据采样插补由粗插补和精插补两个步骤组成。在粗插补阶段(一般数据采样插补都是指粗插补)，是采用时间分割思想，根据编程规定的进给速度  $F$  和插补周期  $T$ ，将廓形曲线分割成一段段的轮廓步长  $l$ ， $l = FT$ ，然后计算出每个插补周期的坐标增量  $\Delta X$  和  $\Delta Y$ ，进而计算出插补点(即动点)的位置坐标。在精插补阶段，要根据位置反馈采样周期的大小，由伺服系统完成。也可以用基准脉冲法进行精插补。

### 2. 插补周期和采样周期

插补周期  $T$  的合理选择是数据采样插补的一个重要问题。在一个插补周期  $T$  内，计算机除了完成插补运算外，还要执行显示、监控和精插补等项实时任务，所以插补周期  $T$  必须大于插补运算时间与完成其它实时任务时间之和，一般为 8 ~ 10 ms 左右，现代数控系统已缩短到 2 ~ 4 ms，有的已达到零点几毫秒。此外，插补周期  $T$  还会对圆弧插补的误差产生影响。

插补周期  $T$  应是位置反馈采样周期的整数倍，该倍数应等于对轮廓步长实时精插补时的插补点数。

### 3. 插补精度分析

(1) 直线插补时，由于坐标轴的脉冲当量很小，再加上位置检测反馈的补偿，可以认为轮廓步长  $l$  与被加工直线重合，不会造成轨迹误差。

(2) 圆弧插补时，一般将轮廓步长  $l$  作为弦线或割线对圆弧进行逼近，因此存在最大半径误差  $e_r$ ，如图 3.12 所示。

采用弦线对圆弧进行逼近时，根据图 3.12a 可知：

$$r^2 - (r - e_r)^2 = \left(\frac{l}{2}\right)^2$$

$$2re_r - e_r^2 = \frac{l^2}{4}$$

舍去高阶无穷小  $e_r^2$ ，则

$$e_r = \frac{l^2}{8r} = \frac{(FT)^2}{8r} \quad (3.20)$$

若采用理想割线、又称内外差分弦对圆弧进行逼近，因为内外差分弦使内外半径的误差  $e_r$  相等，如图 3.12b 所示，则有：

$$\begin{aligned} (r + e_r)^2 - (r - e_r)^2 &= \left(\frac{l}{2}\right)^2 \\ 4re_r &= \frac{l^2}{4} \\ e_r &= \frac{l^2}{16r} = \frac{(FT)^2}{16r} \end{aligned} \quad (3.21)$$

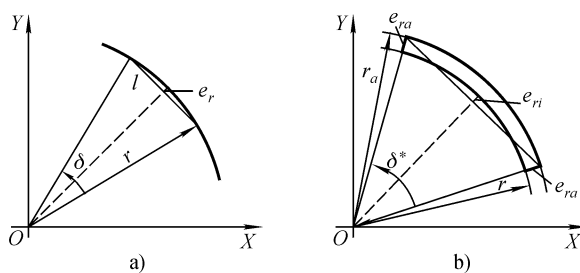


图 3.12 弦线、割线逼近圆弧的径向误差

显然，当轮廓步长相同时，内外差分弦的半径误差是内接弦的一半：若令半径误差相等，则内外差分弦的轮廓步长  $l$  或角步距  $\delta$  可以是内接弦的  $\sqrt{2}$  倍，但由于前者计算复杂，很少应用。

以上分析可知，圆弧插补时的半径误差  $e_r$  与圆弧半径  $r$  成反比，而与插补周期  $T$  和进给速度  $F$  的平方成正比。当  $e_r$  给定时，可根据圆弧半径  $r$  选择插补周期  $T$  和进给速度  $F$ 。

### 3.3.2 数据采样法直线插补

#### 1. 插补计算过程

由图 3.13 所示的直线可以看出，在直线插补过程中，轮廓步长  $l$  及其对应的坐标增量  $\Delta X$ 、 $\Delta Y$  等是固定的，因此直线插补的计算过程可分为插补准备和插补计算两个步骤。

(1) 插补准备 主要是计算轮廓步长  $l = FT$  及其相应的坐标增量，可以采用不同方法计算。

(2) 插补计算 实时计算出各插补周期中的插补点(动点)坐标值。

#### 2. 实用的插补算法

##### (1) 直接函数法

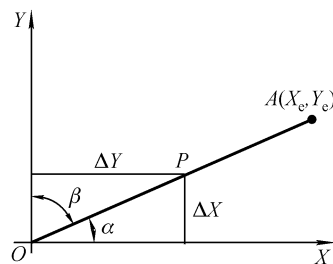


图 3.13 数据采样法直线插补

$$\left. \begin{array}{l} \text{插补准备:} \\ \text{插补计算:} \end{array} \right\} \begin{array}{l} \Delta X_i = \frac{l}{L} X_e \\ \Delta Y_i = \Delta X_i \frac{Y_e}{X_e} \end{array} \quad (3.22)$$

$$X_i = X_{i-1} + \Delta X_i, \quad Y_i = Y_{i-1} + \Delta Y_i \quad (3.23)$$

(2) 进给速率法(扩展 DDA 法)

$$\text{插补准备: 计算步长系数 } K = \frac{l}{L} = \frac{FT}{L} = T \cdot \text{FRN}$$

$$\text{插补计算:} \quad \Delta X_i = K X_e, \quad \Delta Y_i = K Y_e \quad (3.24)$$

$$X_i = X_{i-1} + \Delta X_i, \quad Y_i = Y_{i-1} + \Delta Y_i \quad (3.25)$$

(3) 方向余弦法一

$$\text{插补准备: } \cos \alpha = \frac{X_e}{L}, \quad \cos \beta = \frac{Y_e}{L}$$

$$\text{插补计算:} \quad \Delta X = l \cos \alpha, \quad \Delta Y = l \cos \beta \quad (3.26)$$

$$X_i = X_{i-1} + \Delta X, \quad Y_i = Y_{i-1} + \Delta Y \quad (3.27)$$

(4) 方向余弦法二

$$\text{插补准备: } \cos \alpha = \frac{X_e}{L}, \quad \cos \beta = \frac{Y_e}{L}$$

$$\text{插补计算: } L_i = L_{i-1} + l$$

$$X_i = L_i \cos \alpha, \quad Y_i = L_i \cos \beta \quad (3.28)$$

$$\Delta X_i = X_i - X_{i-1}, \quad \Delta Y_i = Y_i - Y_{i-1} \quad (3.29)$$

(5) 一次计算法

$$\text{插补准备:} \quad \Delta X_i = \frac{l}{L} X_e, \quad \Delta Y_i = \frac{l}{L} Y_e \quad (3.30)$$

$$\text{插补计算:} \quad X_i = X_{i-1} + \Delta X_i, \quad Y_i = Y_{i-1} + \Delta Y_i \quad (3.31)$$

### 3.3.3 数据采样法圆弧插补

由于圆弧是二次曲线，是用弦线或割线进行逼近，因此其插补计算要比直线插补复杂。用直线逼近圆弧的插补算法很多，而且还在发展。研究插补算法遵循的原则：一是算法简单，计算速度快；二是插补误差小，精度高。下面简要介绍日本 FANUC 公司 7 系统采用的直线函数法、美国 A-B 公司采用的扩展 DDA 法以及递归函数法。

#### 1. 直线函数法(弦线法)

在图 3.14 中，顺圆上 B 点是继 A 点之后的瞬时插补点，坐标值分别为  $A(X_i, Y_i)$ 、 $B(X_{i+1}, Y_{i+1})$ 。为了求出 B 点的坐标值，过 A 点作圆弧的切线 AP，M 是弦线 AB 的中点，AF 平行与 X 轴，而 ME、BF

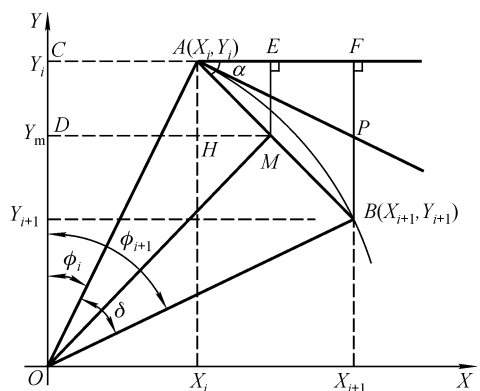


图 3.14 直线函数法圆弧插补

平行于  $Y$  轴。 $\delta$  是轮廓步长  $AB$  弦对应的角步距。 $OM \perp AB$ ,  $ME \perp AF$ ,  $E$  为  $AF$  的中点。  
因为  $OM \perp AB$ ,  $AF \perp OD$

所以 
$$\alpha = \angle MOD = \varphi_i + \frac{\delta}{2}$$

在  $\triangle MOD$  中, 有

$$\tan\left(\varphi_i + \frac{\delta}{2}\right) = \frac{DH + HM}{OC - CD}$$

将  $DH = X_i$ ,  $OC = Y_i$ ,  $HM = \frac{1}{2}l\cos\alpha = \frac{1}{2}\Delta X$  和  $CD = \frac{1}{2}l\sin\alpha = \frac{1}{2}\Delta Y$ , 代入上式, 则有

$$\tan\alpha = \frac{X_i + \frac{1}{2}l\cos\alpha}{Y_i - \frac{1}{2}l\sin\alpha} = \frac{X_i + \frac{1}{2}\Delta X}{Y_i - \frac{1}{2}\Delta Y} \quad (3.32)$$

在上式中,  $\sin\alpha$  和  $\cos\alpha$  都是未知数, 难以用简单方法求解, 因此采用近似计算求解  $\tan\alpha$ , 用  $\cos 45^\circ$  和  $\sin 45^\circ$  来取代, 即

$$\tan\alpha \approx \frac{X_i + \frac{\sqrt{2}}{4}l}{Y_i - \frac{\sqrt{2}}{4}l}$$

从而造成了  $\tan\alpha$  的偏差, 使角  $\alpha$  变为  $\alpha'$  (在  $0 \sim 45^\circ$  间,  $\alpha' < \alpha$ ), 使  $\cos\alpha'$  变大, 因而影响  $\Delta X$  值使之成为  $\Delta X'$ , 即

$$\Delta X' = l\cos\alpha' = AF' \quad (3.33)$$

$\alpha$  角的偏差会造成进给速度的偏差, 而在  $\alpha$  为  $0^\circ$  和  $90^\circ$  附近偏差较大。为使这种偏差不会使插补点离开圆弧轨迹,  $Y'$  不能采用  $l\sin\alpha'$  计算, 而采用式 (3.34) 来计算, 即

$$\Delta Y' = \frac{\left(X_i + \frac{1}{2}\Delta X'\right)\Delta X'}{Y_i - \frac{1}{2}\Delta Y'} \quad (3.34)$$

则  $B$  点一定在圆弧上, 其坐标为

$$X_{i+1} = X_i + \Delta X', \quad Y_{i+1} = Y_i - \Delta Y'$$

采用近似计算引起的偏差仅是  $\Delta X \rightarrow \Delta X'$ ,  $\Delta Y \rightarrow \Delta Y'$ ,  $\Delta l \rightarrow \Delta l'$ 。这种算法能够保证圆弧插补的每一插补点位于圆弧轨迹上, 它仅造成每次插补的轮廓步长、即合成进给量  $l$  的微小变化, 所造成的进给速度误差小于指令速度的 1%, 这种变化在加工中是允许的, 完全可以认为插补的速度仍然是均匀的。

## 2. 扩展 DDA 法数据采样插补

扩展 DDA 算法是在 DDA 积分法的基础上发展起来的, 它是将 DDA 法切线逼近圆弧的方法改变为割线逼近, 从而大大提高圆弧插补的精度。

如图 3.15 所示, 若加工半径为  $R$  的第一象限顺时针圆弧  $AD$ , 圆心为  $O$  点, 设刀具处在现加工点  $A_{i-1}(X_{i-1}, Y_{i-1})$  位置, 线段  $A_{i-1}A_i$  是沿被加工圆弧的切线方向的轮廓进给步长,  $A_{i-1}A_i = l$ 。显然, 刀具进给一个步长后, 点  $A_i$  偏离所要求的圆弧轨迹较远, 径向误差较大。若通过  $A_{i-1}A_i$  线段的中点  $B$ , 作以  $OB$  为半径的圆弧的切线  $BC$ , 并在  $A_iH$  上截取直线段

$A_{i-1}A'_i$ , 使  $A_{i-1}A'_i = A_{i-1}A_i = l = FT$ , 此时可以证明  $A'_i$  点必定在所要求圆弧  $AD$  之外。如果用直线段  $A_{i-1}A'_i$  替代切线  $A_{i-1}A_i$  进给, 会使径向误差大大减小。这种用割线进给代替切线进给的插补算法称为扩展 DDA 算法。

下面推导在一个插补周期  $T$  内, 轮廓步长  $l$  的坐标分量  $\Delta X_i$  和  $\Delta Y_i$ , 因为据此可以很容易求出本次插补后新加工点  $A'_i$  的坐标位置  $(X_i, Y_i)$ 。

由图 3.15 可知, 在直角  $\triangle OPA_{i-1}$  中,

$$\sin \alpha = \frac{OP}{OA_{i-1}} = \frac{X_{i-1}}{R}$$

$$\cos \alpha = \frac{A_{i-1}P}{OA_{i-1}} = \frac{Y_{i-1}}{R}$$

过  $B$  点作  $X$  轴的平行线  $BQ$  交  $Y$  轴于  $Q$ , 并交  $A_{i-1}P$  线段于  $Q'$  点。由图中可知, 直角  $\triangle OQB$  与直角  $\triangle A_{i-1}MA'_i$  相似, 则有

$$\frac{MA'_i}{A_{i-1}A'_i} = \frac{OQ}{OB} \quad (3.35)$$

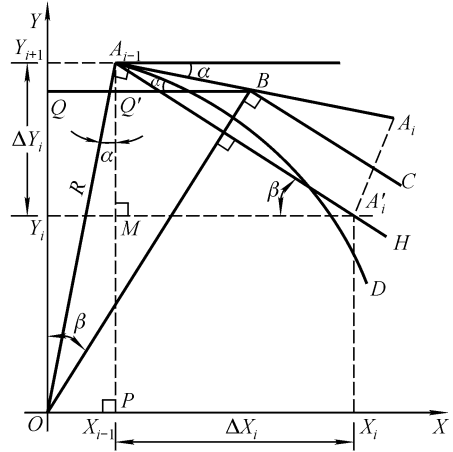


图 3.15 扩展 DDA 法圆弧插补算法

在图 3.15 中,  $MA'_i = \Delta X_i$ ,  $A_{i-1}A'_i = l$ , 在  $\triangle A_{i-1}Q'B$  中,  $A_{i-1}Q' = A_{i-1}B \cdot \sin \alpha = \frac{1}{2}l \cdot \sin \alpha$ , 则

$$OQ = A_{i-1}P - A_{i-1}Q' = Y_{i-1} - \frac{1}{2}l \cdot \sin \alpha$$

在直角  $\triangle OA_{i-1}B$  中

将  $OQ$  和  $OB$  代入式(3-35)中, 得  $OB = \sqrt{(A_{i-1}B)^2 + (OA_{i-1})^2} = \sqrt{\left(\frac{1}{2}l\right)^2 + R^2}$

$$\frac{\Delta X_i}{l} = \frac{Y_{i-1} - \frac{1}{2}l \sin \alpha}{\sqrt{\left(\frac{l}{2}\right)^2 + R^2}}$$

上式中, 因为  $l \ll R$ , 故可将  $\left(\frac{1}{2}l\right)^2$  略去, 则上式变为

$$\Delta X_i \approx \frac{l}{R} \left( Y_{i-1} - \frac{1}{2}l \frac{X_{i-1}}{R} \right) = \frac{FT}{R} \left( Y_{i-1} - \frac{1}{2} \cdot \frac{FT}{R} X_{i-1} \right) \quad (3.36)$$

在相似直角  $\triangle OQB$  与直角  $\triangle A_{i-1}MA'_i$  中, 还有

$$\frac{A_i M}{A_{i-1}A'_i} = \frac{QB}{OB} = \frac{QQ' + Q'B}{OB}$$

在直角  $\triangle A_{i-1}Q'B$  中, 有  $Q'B = A_{i-1}B \cdot \cos \alpha = \frac{l}{2} \cdot \frac{Y_{i-1}}{R}$ , 又  $QQ' = X_{i-1}$ , 则

$$\Delta Y_i = A_{i-1}M = \frac{A_{i-1}A'_i (QQ' + Q'B)}{OB} = \frac{l \left( X_{i-1} + \frac{1}{2}l \frac{Y_{i-1}}{R} \right)}{\sqrt{\left(\frac{l}{2}\right)^2 + R^2}}$$

同理，由于  $l \ll R$ ，略去高阶无穷小  $\left(\frac{l}{2}\right)^2$ ，则有

$$\Delta Y_i \approx \frac{l}{R} \left( X_{i-1} + \frac{1}{2} \cdot \frac{l}{R} Y_{i-1} \right) = \frac{FT}{R} \left( X_{i-1} + \frac{1}{2} \cdot \frac{FT}{R} Y_{i-1} \right) \quad (3.37)$$

若令  $K = \frac{FT}{R} = T \cdot \text{FRN}$ ，则

$$\begin{cases} \Delta X_i = K \left( Y_{i-1} - \frac{1}{2} K X_{i-1} \right) \\ \Delta Y_i = K \left( X_{i-1} + \frac{1}{2} K Y_{i-1} \right) \end{cases} \quad (3.38)$$

则  $A'_i$  点的坐标值为

$$\begin{cases} X_i = X_{i-1} + \Delta X_i \\ Y_i = Y_{i-1} - \Delta Y_i \end{cases} \quad (3.39)$$

式(3.38)和(3.39)为第一象限顺时针圆弧插补计算公式，依照此原理，可得出其它象限及其走向的扩展 DDA 圆弧插补计算公式。

由上述扩展 DDA 圆弧插补公式可知，采用该方法只需进行加法、减法及有限次的乘法运算，因而计算较方便、速度较高。此外，该法用割线逼近圆弧，其精度较弦线法高。因此扩展 DDA 法是比较适合于 CNC 系统的一种插补算法。

### 3. 递归函数算法(RFB)

递归函数采样插补是通过轨迹曲线参数方程的递归计算实现插补的。由于它是根据前一个或前两个已知插补点来计算本次插补点，故称为一阶递归插补或二阶递归插补。

#### 1) 一阶递归插补

图 3.16 为要插补的圆弧，起点为  $P_0(X_0, Y_0)$ ，终点为  $P_E(X_E, Y_E)$ ，圆弧半径为  $R$ ，圆心位于坐标原点，进给速度为  $F$ 。设刀具现在位置为  $P_i(X_i, Y_i)$ ，经过一个插补周期  $T$  后到达  $P_{i+1}(X_{i+1}, Y_{i+1})$ ，刀具运动轨迹为  $P_i P_{i+1}$ ，每次插补所转过的圆心角为  $\theta$ ，称为步距角， $\theta \approx \frac{FT}{R} = K$ 。有

$$\begin{cases} X_i = R \cos \varphi_i \\ Y_i = R \sin \varphi_i \end{cases}$$

插补一步后，有  $\varphi_{i+1} = \varphi_i - \theta$

$$\begin{cases} X_{i+1} = X_i \cos \theta + Y_i \sin \theta \\ Y_{i+1} = X_i \sin \theta - Y_i \cos \theta \end{cases} \quad (3.40)$$

式(3.40)称为一阶递归插补公式。

将式(3.40)中的三角函数  $\cos \theta$  和  $\sin \theta$  用幂级数展开进行二阶近似，即

$$\cos \theta \approx 1 - \frac{\theta^2}{2} = 1 - \frac{K^2}{2}$$

$$\sin \theta \approx \theta \approx K$$

代入式(3.40)，则

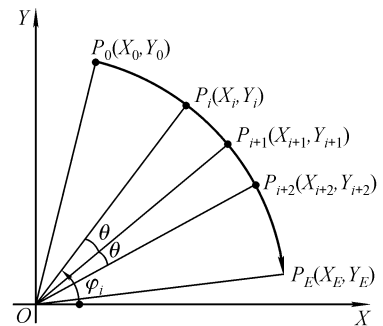


图 3.16 函数递归法圆弧插补

$$\begin{cases} X_{i+1} = X_i + K \left( Y_i - \frac{1}{2} K X_i \right) \\ Y_{i+1} = Y_i - K \left( X_i + \frac{1}{2} K Y_i \right) \end{cases}$$

这个结果与扩展 DDA 法插补的结果一致，因此扩展 DDA 法也可称为一阶递归二阶近似插补。

## 2) 二阶递归插补

二阶递归插补算法中，需要两个已知插补点。若插补点  $P_{i+1}$  已知，则对于下一插补点  $P_{i+2}$  有  $\varphi_{i+2} = \varphi_{i+1} - \theta$ ，则

$$\begin{cases} X_{i+2} = X_{i+1} \cos \theta + Y_{i+1} \sin \theta \\ Y_{i+2} = X_{i+1} \sin \theta - Y_{i+1} \cos \theta \end{cases} \quad (3.41)$$

将式(3-40)代入式(3-41)，则有

$$\begin{cases} X_{i+2} = X_i + 2Y_{i+1} \sin \theta = X_i + 2Y_{i+1} K \\ Y_{i+2} = Y_i - 2X_{i+1} \cos \theta = Y_i - 2X_{i+1} K \end{cases} \quad (3.42)$$

显然计算更为简单。但二阶递归插补需要用其它插补法计算出第二个已知的插补点  $P_{i+1}$ ，同时考虑到误差的累积影响，参与计算的已知插补点应计算得尽量精确。

## 3.4 曲面直接插补(SDI)

实现高速高精度加工一直是数控技术研究的重点，但目前多数 CNC 系统在轨迹控制上依然只有直线、圆弧等少数功能，在曲面加工中，要形成十分庞大的由微段直线构成的零件程序，即使使用先进的 CAD/CAM 编程功能，其程序制作的时间也往往是实际机床加工时间数倍乃至数十倍，代价十分昂贵。庞大的零件程序是妨碍 CAD/CAM 工作效率的“瓶颈”。由于精加工程序的形成代价过高，目前的曲面数控加工，除关键零件外，多数只作为粗成型使用，然后由人工打磨修光，不仅效率低，而且难以保证型面精度。

所谓曲面直接插补(Surface Direct Interpolation, 简称 SDI)是将曲面加工中的刀具运动轨迹产生功能集中到 CNC 中，由 CNC 直接根据曲面定义和工艺参数，实时地自动完成连续刀具轨迹插补，并由此来控制机床运动。应用曲面直接插补技术，可大大简化编程工作与加工信息，不仅对 CNC 系统，而且对 CAD/CAM 整体性能的提高也极为关键。日、美、德等国对此作了大量研究，FANUC 15M 等扩充了三次样条曲线插补功能，SIEMENS840D 具有了 A、B、C 三种样条曲线插补功能，美国 NGC 计划中也将样条曲线作为重要功能之一。但对曲面实时加工的实用化系统迄今为止还在研究和完善中。华中科技大学完成了曲面直接插补的技术开发，下面对该项技术作简要介绍。

### 3.4.1 SDI 的优点

1. 在 CNC 系统中实现曲面加工的连续刀具轨迹直接插补，正如一般 CNC 系统中有圆弧功能可对圆弧直接加工一样，使得工程曲面也成为 CNC 系统的内部功能而直接调用，大大简化了编程。

2. 在轨迹插补中采用 CNC 系统周期所决定的进给步长直接逼近，其步长只取决于加工速度和系统插补周期，可以获得最高加工精度。而目前 CNC 的曲面加工精度是由 CAM 的编程周期所决定，为了避免编程信息量过于庞大，CAM 的编程步长不能过小，从而限制了加工精度的提高。

3. 由于曲面插补是在 CNC 内部进行，避免了由 CAM 生成的连续微段程序的传输过程，从而可以实现高速加工运动的控制，使 CNC 系统具有高速、高精加工能力，可大大提高加工效率。

4. 在轨迹插补中考虑了刀具与加工余量的补偿，因此可由操作者根据实际加工情况对工艺参数等进行现场修改、调整，以适应具体的加工零件，发挥操作者的丰富经验，可以大大提高加工的灵活性。

### 3.4.2 SDI 的功能与信息输入

由于曲面加工的复杂性，零件的 SDI 程序必须使用类似于 APT 级的直接面向加工问题的高级语言，不仅要针对轨迹上的进给分量进行脉冲分配，还要包括工艺参数(刀具、加工余量)和加工过程(加工路线、进退刀)的综合表达，以便完成一系列的机床运动轨迹插补、行距判别、辅助进退刀处理等，即完成类似 CAM 的功能。其主要功能如下：

#### 1. 线加工

线加工是曲面加工的基础，用依次排列的行切曲线加工实现曲面加工，是曲面加工中最简洁的方法。在光滑且无干涉的曲面  $S_i$  上，给定曲面曲线的起点( $P_0$ )、终点( $P_e$ )，则 CNC 能根据所用刀具与加工余量，沿行切方向实现相应刀具轨迹插补，实现机床加工运动控制。若将曲面上的一系列列表分界点作为加工序列  $[P_0, P_e]$ ，辅以适当的进刀和退刀控制并加以组合，就能实现整个区域加工，如图 3.17a 所示。其加工语句为

$$\text{Path} = S_i / [P_0, P_e]$$

#### 2. 区域加工

在由对角点( $P_0, P_e$ )给出的矩形域曲面  $S_i$  上，SDI 除完成当前刀具轨迹插补外，还要根据表面允许残余高度自动进行下一步走刀的行距确定，并按指定的加工路线方式，完成相应的辅助进、退刀运动，从而自动完成整张曲面的加工，如图 3.17b 所示。其加工语句为

$$\text{Surf} = S_i / [P_0, P_e], \text{Path-type};$$

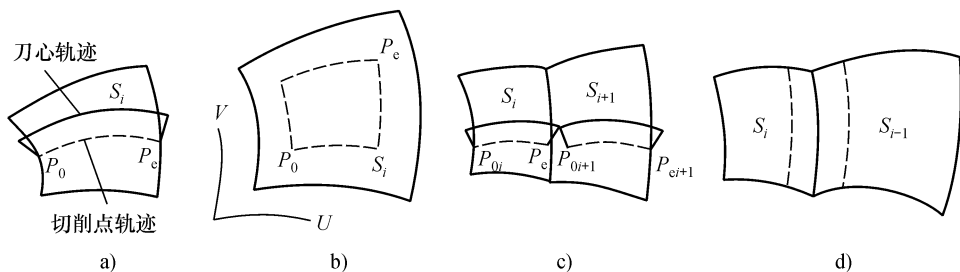


图 3.17 曲面加工方式

其中 Path-type 为加工路线方式，共提供三种走刀方式。zig-zag(双向来回走刀)，zig(单向走

刀), close(闭合走刀), 缺省方式为 zig-zag。在上述定义中, SDI 使用了类似于 APT 系统中的加工语句, 其区域加工的表达, 除干涉处理外, 已与大型的 CAM 的实现内容相同; 线加工的表达式, 在形式上也类似于 APT 系统中的轮廓加工语句, 其中  $S_i$  为零件面(即控制刀具高度的表面),  $P_0, P_e$  类似于导动面(即控制运动方向的表面), 以两点形式代替原来的离散直线。

### 3. 组合曲面加工

对于复杂组合曲面, 可由 Surf 功能逐片进行加工, 完成大部分加工任务, 再在边界上按标准 NC 功能完成干涉区加工, 如图 3.17c 所示, 也可按加工方式进行连续加工, 如图 3.17d 所示。

### 3.4.3 SDI 的结构和工作流程

SDI 插补器的结构和工作流程如图 3.18 所示。各功能模块说明如下。

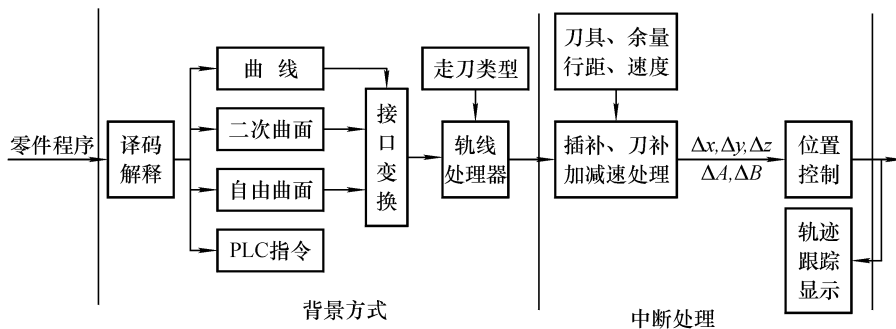


图 3.18 SDI 工作流程和基本结构

1. 译码解释, 根据零件程序, 进行译码解释和组合曲面的预处理。
2. 将各类双三次自由曲面归于标准双三次函数, 以统一的插补方式工作。若为二次曲面, 则将其归于标准二次型方程工作。
3. 轨线处理, 按照给定的加工方式(单向、双向、闭合)和允许的残余高度, 实时处理出走刀路线和辅助进、退刀路线。
4. 插补、刀补和加减速处理, 根据走刀路线、刀具形状尺寸、加工余量、非线性修调及五轴机床摆动机构, 实时计算出运动控制轨迹, 并在速度转接处和轨迹起、终点处自动加、减速处理。
5. 跟踪显示, 将当前机床运动位置以三维方式进行跟踪显示, 实现加工过程的监视。

### 3.4.4 SDI 的算法原理

在 CNC 上实现 SDI 功能, 核心问题在于插补器的实现, 内容包括刀具轨迹的插补, 机床加减速控制、随加工型面变化的速度修调以及切削行宽度的确定等多项功能。

#### 1. 刀具轨迹插补

刀具轨迹插补是 SDI 的工作核心, 由曲面给出定义, 然后根据刀具形状和加工余量, 按指定的进给速度, 在规定的插补周期内, 实时地计算出各坐标轴的运动分量, 作为伺服驱动的速

度指令。

对于被加工的自由曲面，一般用参数表达式定义，对于各种参数曲面均可表达为

$$\mathbf{r}_s = \mathbf{r}_s(u, v) = \sum_{i=0}^m \sum_{j=0}^n x_{ik}(u) x_{jk}(v) Q_{ij}$$

其中  $u, v$  是形成自由曲面的两个方向上的参数，其取值范围为  $0 \leq u, v \leq 1$ ， $Q_{ij}$  是已知的几何条件，可以理解为对曲面进行控制的型值点的矢量(坐标)； $x_{ik}(u)$  和  $x_{jk}(v)$  是基函数，改变基函数，可分别表示 b-Spline、Bezier 等不同自由曲面， $k$  为曲面阶次，工程实用中取  $k=3$ 。

三坐标以上曲面加工一般使用球头铣刀，也可使用环形、鼓形或平头刀具，这些刀具均可用半径  $R_1$  和  $R_2$  来表示，其与零件面的接触关系如图 3.19 所示。设曲面单位法矢为  $\mathbf{n}$ ，加工余量用矢量式表示为  $\mathbf{r}_\delta = b\mathbf{n}$ ，刀具长度补偿用矢量式表示为  $\mathbf{r}_L = (0, 0, \Delta L)$ ，则刀具在工件坐标系下的运动轨迹可表示为

$$\mathbf{r}_m(t) = \mathbf{r}(u(t), v(t)) + \mathbf{r}_\delta(t) + R_1 \mathbf{n} + O_1 O_2 + \mathbf{r}_L(t)$$

轨迹插补的任务即是在时间  $t = kT$  ( $T$  为插补周期,  $k$  为插补周期序号) 时，按指定进给速度  $F(t)$ ，实时地计算出机床运动分量，以满足

$$\Delta \mathbf{r}_m(kT) = \mathbf{r}_m(kT) - \mathbf{r}_m((k-1)T)$$

一般情况下，总希望刀具相对加工表面成恒速关系。但三坐标加工时，随着型面变化，刀具在各切削点的切削情况不同，因此保持恒表面切削不易实现，故而按标准 NC 方式使刀具中心作恒速运动，则机床运动规律为

$$\Delta \mathbf{r}_m(t) = F(t)$$

## 2. 机床加减速控制

为了在机床启动、停止、轨迹转折、速度变化处保证平滑过渡，必须使机床按给定平滑规律进行加减速处理。SDI 采用了常用的线性和指数加减速规律。在切削加工中，使用平滑性好的指数规律，在辅助退刀时，采用快速性好的直线规律。

设机床进给系统的时间常数为  $\tau$ ，则正常时  $F(t) = F_0$ ，按指数规律加减速时为

$$F(t) = \begin{cases} F_0 \left( 1 - e^{-\frac{t}{\tau}} \right) & \text{加速时} \\ F_0 e^{-\frac{t}{\tau}} & \text{减速时} \end{cases}$$

在加减速控制中，速度变化采用递推运算，在加速时速度依次递增即可，但在减速时，减速区的判断较为困难，需重点解决。

在加减速控制方式中，有插补前加减速和插补后加减速两种方式。其中插补前加减速沿轨迹方向上对速度进行控制，不会造成轨迹误差，但需较复杂的沿弧长方向上的路径计算；而插补后加减速方式则根据各轴到终点的坐标方向上的差值、通过改变系统回路增益来控制，其减速区计算简单，但会由于机床各坐标轴伺服特性不一致而形成轨迹误差。

由于刀具轨迹是空间曲线，计算空间曲线弧长较为困难。而采用按各坐标轴与终点距离的坐标进行判别时，则会由于曲线凹凸不定，造成误判别，虽然可将其分段处理，但会给 SDI 解释和 CAM 处理造成困难。因此 SDI 采用了沿轨迹方向上的加减速控制方式，采用快速数值积

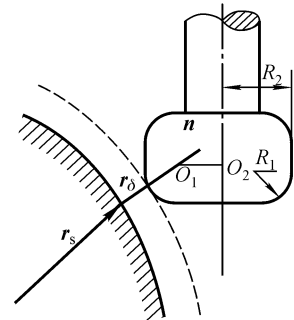


图 3.19 刀具与零件面的接触关系

分来计算至终点的弧长来进行减速区判别，从而实现了高精度的插补前加减速控制方式。

此外，在一般的 CNC 系统中，都是假定当前程序段较长，能有充分的减速时间，但如果当前程序段较短且机床运动速度很高时，则有可能还未达到所要求的速度就已超过减速区。为此，SDI 在插补过程中，处处根据当前速度进行减速区域监视，以保证具有足够的减速区域。

### 3. 随加工型面曲率变化的速度修调

为了提高加工精度，在凹曲面加工中，SDI 根据型面曲率变化进行了速度修调。

凹曲面加工时，刀具中心轨迹的曲率半径为型面曲率半径  $\rho$  与刀具半径  $R$  之差，当两者相差很小时，若仍按正常进给速度  $F$  进行进给，则会造成较大的径向误差。此外由于刀具中心轨迹曲率半径过小，会造成速度方向变化过大和机床惯性导致型面成为多角形，其加工情况如图 3.20 所示。设径向误差为  $\delta$ ，刀具进给的步距角为  $\theta$ ，则有

$$\delta = (\rho - R) \left( 1 - \cos \frac{\theta}{2} \right)$$

$$\theta = \arcsin \frac{FT}{\rho - R}$$

当  $\rho \rightarrow R$  时，步距角  $\theta$  增加，导致径向误差  $\delta$  增大，因此 SDI 必须将进给速度  $F$  修调为  $F_1$ ，

$$F_1 = \begin{cases} F \cdot \frac{(\rho - R)}{\rho} & \rho > R \\ 0 & \rho \leq R \end{cases}$$

当  $\rho \leq R$  时，已经出现加工干涉，此时应该停机，更换小直径刀具。当  $\rho > R$  时，通过修调进给速度  $F$ ，使刀具进给步距角减少为

$$\theta = 2 \arcsin \frac{F_1 T}{2(\rho - R)} \approx \frac{FT}{\rho}$$

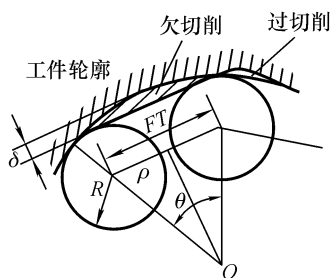


图 3.20 凹曲面曲率较大时的加工情况

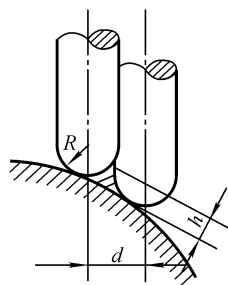


图 3.21 行间残留高度

此时的逼近误差为

$$\delta = (\rho - R) \left( 1 - \cos \frac{\theta}{2} \right) \approx (\rho - R) \frac{\theta^2}{8} \approx (\rho - R) \frac{(FT)^2}{8\rho^2}$$

在加工凸曲面时，由于刀具中心轨迹的曲率半径为零件曲率半径和刀具半径之和，较为平坦，为提高实时性，SDI 不对进给速度进行修调。

### 4. 切削行宽度的确定

切削行宽度  $d$  较大时，会提高加工效率，但会增加切削残留高度  $h$ ，如图 3.21 所示，因

此必须合理确定切削行宽度，根据两条曲面曲线所要求的最小距离来确定。但严格的计算要涉及曲面形状，计算量很大，意义也不大。因此 SDI 采用许多 CAM 采用的简化算法，将行间局部区域以平面近似代替曲面。则

$$d = 2 \sqrt{h(2R - h)}$$

在精加工时，由于行间距不大，一般可获得满意的结果。为能按轨迹全长考虑，SDI 中使用了 32 ms 为中断周期的行距监视，以获得行方向全长上的最小增量控制。

### 3.4.5 SDI 的技术关键

SDI 的工作内容类似于 CAM，其曲面块加工指令相当于一个完整工步，由于 SDI 工作在 CNC 实时环境下，其控制复杂度远远超过目前 CNC 的直线和圆弧处理，除了要解决前面论述的有关算法外，还要解决下面两个技术关键。

#### 1. 插补器的实时处理速度

CNC 是实时性系统。由于曲面加工的轨迹处理要比直线和圆弧计算复杂得多，要在几毫秒内完成整个计算，仅仅依靠提高硬件速度是十分有限的，必须通过有效的算法提高计算速度。

#### 2. 插补器稳定性问题

插补算法除能实现高速处理外，还须充分考虑其稳定性。因为计算机字长有限，复杂轨迹的计算需以浮点进行，且其迭代次数巨大。如插补周期为 8 ms 时，一小时加工需要 45 万次运算。要在如此巨大的迭代次数中不产生累积误差，其数值稳定性是算法和软件设计中的又一关键。

## 复 习 题

1. 何谓插补？数控加工为什么要使用插补？
2. 有哪两类插补算法？它们各有什么特点？
3. 逐点比较法是如何实现的？
4. 试述 DDA 插补原理？
5. 数据采样插补是如何实现的？
6. 数据采样直线插补、圆弧插补有否误差？数据采样插补误差与什么有关系？
7. 如何实现数控系统中没有相对应指令曲线的插补？
8. 高速插补有何特点？如何实现高速插补？
9. 可否用数学函数式直接作为插补算法公式？为什么？
10. 曲面直接插补的原理是什么？