

# 基于知识的装配前趋约束图的生成\*

陈正 张铤

(清华大学计算机科学技术系 北京 100084)

(清华大学智能技术与系统国家重点实验室 北京 100084)

E-mail: snow@sunny.bjnet.edu.cn

**摘要** 装配前趋约束图的生成是装配规划的关键环节,也是其中最困难的一步。该文在讨论了几种装配前趋约束图生成方法的具体实现以及它们的优缺点之后,提出利用几何知识和装配经验知识相结合的方法,建立装配知识库/数据库,给出了几种在复杂度上可行的解决方法。最后给出结论,并简单地讨论了结合知识的数学算法的优越性,同时讨论了今后发展的方向。

**关键词** 自动装配,阻元,前趋约束图,几何知识,三角模型,装配数据库/知识库。

**中图法分类号** TP391

随着科学技术的高度发展,工业上需求的装配渐渐向小批量、短周期、低成本的方向发展。<sup>[1]</sup>这就必然要求装配计划的制定(即装配规划)能适应这些要求,因此,装配规划在性能、速度和自动化程度方面都要大大提高。

在装配规划中,装配序列规划是实现装配机器人任务级编程的核心技术之一。实现任务级编程是提高机器人编程效率和灵活性的重要途径。装配序列规划在任务级编程系统中占有重要的地位,它不仅直接影响机器人的“粗”运动规划和“精”运动规划,而且还影响到装配工具和装配方式的选择以及零件的抓取规划。对装配序列规划方法的研究开始于80年代初期,如 Wesley 等人在1980年和 Takeyama 等人在1983年的工作,到了80年代后期才引起广泛重视。它的表示方法可以分为两种:(1)基于约束的表示方法;(2)基于枚举的表示方法。当序列集增大时,基于约束表示的序列集只是缓慢增大,而基于枚举表示的序列集迅速增大。常见的基于约束的装配序列表示是前趋关系。

装配序列规划的方法基本上是采用“变装为拆”的方法,具体实现上可分为两类:一类是穷举搜索法,如 Homen de Mello 和 Sanderson, DeFazio 和 Whitney<sup>[2]</sup>, Baldwin 等人的研究。<sup>[3]</sup>另一类以有限可行的装配序列为目标,借助装配序列前趋约束图,采用启发式搜索,如 Huang 和 Lee<sup>[4]</sup>, Delchambre<sup>[5]</sup>等人的研究。前一种算法复杂度较高,后者则有较好的实用性,但通用性较差。

在装配序列规划中,前趋约束图的生成是装配序列规划的关键环节,也是其中最困难的一步。在装配规划的最前端,规划器接收来自工程人员设计的CAD图纸,规划出相应的、能让机械手装配的序列。我们知道,在目前普遍使用的几何造型系统中,零件的表示方法主要有 CSG 和 B-rep 两种。但是对于规划系统,这些几何信息很难使用,原因在于这些数据量庞大,而且不容易找出其中各个零件的关系。因此,把零件的几何信息转换成规划器能接收的数据格式,就成为必不可少的一个环节,也成为联接人和机器规划的桥梁和关键。此外,在装配规划中仅仅使用到零件的几何信息还是很不够的,需要通过描述零件的装配特征来达到简化装配复杂度和提高装配速度的目的。

为了生成满足各种装配约束的装配规划序列,我们需要从装配模型中提取出生成装配序列所需要的信息,在装配序列规划阶段,忽略一些不必要的信息,把重点放在决定装配序列的各个零件之间的几何和工艺约束上。而前趋约束关系正是符合这个条件,用来表示零件之间的约束关系。

智能装配机器人序列规划中的装配件的前趋约束关系的生成方法,大体上可以分为以下几个部分:(a)根据装配件的约束(匹配)关系。(b)根据装配件的几何信息。(c)前述(1)和(2)两种方法的结合。

下面就这些方法作具体的介绍。

\* 本文研究得到国家自然科学基金和国家 863 高科技项目基金资助。作者陈正,1972年生,博士,主要研究领域为计算机应用技术,人工智能。张铤,1935年生,教授,博士生导师,中国科学院院士,主要研究领域为计算机应用技术,人工智能。

本文通讯联系人:陈正,北京 100084,清华大学计算机科学技术系

本文 1997-05-19 收到原稿,1997-06-25 收到修改稿

### 1 一些基本概念

在机械装配过程中,每个零件都是沿着初始的装配轨迹运动到目标位置,若移动过程中不和其他零件发生碰撞,我们则称这样的运动轨迹为该零件的(可行)装配路径。

一般来说,零件的装配路径可以表示成三维空间的任意一条曲线,记为单位向量  $\tau \in [t, \omega] = R^6$ 。当沿着装配路径  $\tau$  装配这个零件,该零件可以沿着  $t$  方向平移,同时绕  $\omega$  作旋转。但是,在实际装配过程中,装配路径多是沿着直线运动,因此,  $\|\omega\| = 0$ ,而且装配路径的数目也是有限的,这主要是受到了装配机器人的限制,因此,本文讨论的都是基于装配路径是直线的情况。

在介绍前趋约束关系之前,我们先引入一个概念,就是阻元。

设装配件由  $N$  个零件组成,  $P = \{p_i | i = 1, 2, \dots, k \leq N\}$  是零件的集合,  $R = \{q_i | i = 1, 2, \dots, M\}$  是分装件的集合,其中每个  $q_i$  是包含于  $P$  的集合,也就是说,每个分装件是由零件组成的,  $2 \leq \|q_i\| \leq N$ ,  $T = \{t_i | i = 1, 2, \dots, l\}$  为装配路径集合。

定义 1.1. 设  $p, q \in P \cup R, \tau \in T$ , 我们称  $p$  是  $q$  的显阻元,若满足以下条件:

- (1) 当零件  $p$  已经到达目标位置时,它占有的空间是  $V(p)$ ;
- (2) 当零件  $q$  沿着装配路径  $\tau$  从很远的地方向目标位置运动时,它扫描过的空间记为  $Sweep(q, \tau)$ ;
- (3) 如果  $V(p) \cap Sweep(q, \tau) \neq \emptyset$ ,

我们记为  $q \xrightarrow{\tau} p$ 。

根据定义,我们知道这个二元关系不是半序关系,它满足自反性,但是不满足传递性。例如,在图 1 中,  $a \xrightarrow{\tau} b$  而且  $b \xrightarrow{\tau} c$ ,但是根据定义,  $a \xrightarrow{\tau} c$  并不成立,也就是说,可以在装  $a$  前装  $c$ ,但是在实际装配过程中却不能先装  $c$  再装  $a$ 。因为如果先装了  $c$ ,那么根据  $b \xrightarrow{\tau} c$ ,  $b$  就无法装上了。因此,我们要推广显阻元的概念,使其满足传递性。

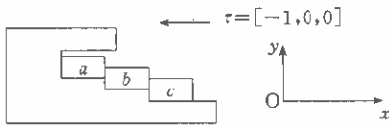


图1 二元关系  $\xrightarrow{\tau}$  不满足传递性

定义 1.2. 设  $p, q \in P \cup R, \tau \in T$ , 我们称  $p$  是  $q$  的隐阻元,若满足以下条件:如果存在  $S = \{s_i | i = 1, 2, \dots, k, k \geq 1\}$ ,使得  $q \xrightarrow{\tau} s_1, s_1 \xrightarrow{\tau} s_2, \dots, s_k \xrightarrow{\tau} p$ , 我们记为  $q \xrightarrow{\tau} p$ 。

推广之后的阻元概念符合半序关系,也就是符合:(1) 自反性;(2) 非对称性;(3) 传递性。

定义 1.3. 设  $p, q \in P \cup R, \tau \in T$ , 我们称  $p$  是  $q$  的阻元,要满足以下任何一个条件之一:(1)  $q \xrightarrow{\tau} p$ ;(2)  $q \xrightarrow{\tau} p$ 。

如何计算两个零件的阻碍关系,我们将在下面的章节中描述,这里,我们先对前趋约束有向图 LCD(local constraint digraph)下一个定义。

定义 1.4. 所谓前趋约束有向图(LCD)就是表示各个零件之间的沿着装配路径的阻碍关系的约束图,记为  $LCD = \{lcd(i, j) | 1 \leq i, j \leq N\}$ ,其中  $lcd(i, j) = 0/1, 0$  表示  $p_i$  和  $p_j$  沿着装配路径  $\tau$  不发生碰撞,1 则表示沿着装配路径  $\tau$  装配的时候,  $p_i$  阻碍了  $p_j$ ,也就是说,  $p_i$  是  $p_j$  的阻元。

由于在计算过程中计算出的两个零件之间的约束关系都是“显阻元”关系,因此,我们需要把由显阻元构成的约束关系图转换成由阻元构成的约束关系图。我们采用的算法是 Warshall 算法。

### 2 基于装配数据库/知识库的算法

在装配中常用的知识主要是指装配零件之间的约束关系。这里说的约束关系主要是指匹配特征关系,如螺母、键、孔等特征,这些特征在装配过程中起着十分重要的作用。因为这些匹配关系决定了这些零件之间的装配方法,大多是只能利用某个特殊的装配方法才能装配的。同时,整个装配件的特征也是十分重要的,例如基础部件等。下面,我们对如何利用装配件的特征作一简单的介绍。

在装配过程中,一些常用的知识是在装配领域中长期积累下来的许多经验知识,如:基本部件可以先从装配序列中去除等。我们在装配中经常能碰到这样的物体,它在整个装配体中属于基本部件,其他零件都或者直接或者间接地和它相连,也就是说,在装配过程中,这个物体是最先装配的,例如某个装配体的底座等。因此,为了让我们的规划程序的复杂性降低,我们把基础部件先抽取出来,对剩余的物体求它们的 LCD 图,装配的时候再把基础部件加上。当然,基础部件的提取取决于经验知识。在装配过程中还存在很多类似的经验知识,他们来自装配过程,又回到实际中为装配服务;在实际装配中也可以积累许多有用的知识并存入知识库,来指导其他装配。

除了一些装配经验知识以外,另外一类很重要的知识就是装配零件的匹配关系.这些匹配关系决定了各个零件之间的几何关系,在CAD中通常使用CSG方法来表示零件之间的约束关系,如某个零件的通孔、螺纹等特征.因此,我们需要建立相应的数据库和知识库来存储相应装配件的特征.在装配过程中,我们就可以利用这些零件特征来求出各个零件的匹配约束关系,继而得到装配件的前趋约束图.下面就比较常见的零件特征作一个介绍.

由于零件的特征很多,如果对所有的特征都一一处理,那么系统的知识库将十分庞大,而且实际装配中也经常遇到一些特殊的零件,这些零件的一些特征在知识库中是不存在的,因此,我们对一些常用的特征进行归纳,利用BNF范式来定义零件特征.

#### 零件表达式

```

<零件参数> ::= (<零件基本参数>, <零件特殊参数>)
<零件基本参数> ::= (<零件名>, <图号>, <重量>, <材料名>, <零件类别>)
<零件特殊参数> ::= (<回转件特殊参数> | <非回转件特殊参数>)
<回转件特殊参数> ::= (<最大回转直径>, 「<最小回转直径>», <最大长度>, <总体特征>, <关键面表达式>, <端面表达式集>)
<零件名> ::= 字符串
<图号> ::= 字符串
<重量> ::= 正实数
<材料名> ::= 钢(STEEL) | 铝(ALUMINIUM) | 铁(IRON) | 其他(OTHERS)
<零件类别> ::= (<回转类> | <非回转类>)
<回转类> ::= WPC /* 轮盘类 */
              | RSC /* 环套类 */
              | AXIS /* 轴类 */
              | PBC /* 销杆类 */
              | GEAR /* 齿轮类 */
              | NUT /* 螺母类 */
              | Other-C /* 其他回转类 */
<非回转类> ::= KEY /* 键类 */
              | Other-NC /* 其他非回转类 */ (对于非回转类,我们研究比较少)
<最大回转直径> ::= 正实数
<最小回转直径> ::= 正实数
<最大长度> ::= 正实数
<总体特征> ::= (<外部总体特征>, 「<内部总体特征>」)
<外部总体特征> ::= SMOOTH /* 沿全长有同一直径外回转表面的零件,
                          对各种螺纹、齿形等结构要素仍算光滑 */
              | SDS /* 单向台阶 */
              | BDS /* 双向台阶 */
              | REPO /* 正多棱柱 */
<内部总体特征> ::= NAH /* 无轴线孔 */
              | SMHO /* 光滑孔 */
              | SDSH /* 单向台阶孔 */
              | BDSH /* 双向台阶孔 */
              | OSBH /* 单侧盲孔 */
              | TSBH /* 双侧盲孔 */
<关键面表达式> ::= (<外部关键面>, 「<内部关键面>」, <外部关键面特征集>, 「<内部关键面特征集>」)
<外部关键面> ::= (<外部关键面名> | <外部关键面名>, <外部关键面>)
<内部关键面> ::= (<内部关键面名> | <内部关键面名>, <内部关键面>)
<外部关键面名> ::= OCYL[<i>] /* 外圆柱面 */
              | OCONE[<i>] /* 外圆锥面 */
              | OCOGEAR[<i>] /* 锥形齿轮面 */
              | OYLGear[<i>] /* 柱形齿轮面 */
              | OPOLY[<i>] /* 正多棱柱面 */

```

```

| ATA[⟨i⟩]          /* 平键表面 */
⟨内部关键面名⟩::= ICYL[⟨i⟩]          /* 内圆柱孔面 */
| ICONE[⟨i⟩]        /* 内圆锥孔面 */
⟨i⟩::= 自然数
⟨外部关键面特征集⟩::= (⟨外部关键面特征⟩)|(⟨外部关键面特征⟩,⟨外部关键面特征集⟩)
⟨内部关键面特征集⟩::= (⟨内部关键面特征⟩)|(⟨内部关键面特征⟩,⟨内部关键面特征集⟩)
⟨外部关键面特征⟩::= ⟨OCYL          [⟨i⟩]⟨C⟩,⟨D⟩,⟨L⟩⟩
| OCYL[⟨i⟩]          (A,⟨D ei es⟩,⟨L⟩)
| OCONE[⟨i⟩]         (C,⟨D1⟩,⟨D2⟩,⟨L⟩)
| OCONE[⟨i⟩]         (A,⟨D1 ei es⟩,⟨D2 ei es⟩⟨L⟩)
| OCOGEAR[⟨i⟩]      (A,⟨D0⟩,⟨L⟩,⟨D⟩,⟨N⟩)
| OYLGEAR[⟨i⟩]     (A,⟨D0⟩,⟨L⟩,⟨N⟩)
| OPOLY[⟨i⟩]        (C,⟨D⟩,⟨L⟩,⟨N⟩)
| ATA              (A,⟨D ei es⟩,⟨L ei es⟩)
⟨内部关键面特征⟩::= ⟨ICYL[⟨i⟩]      (C,⟨D⟩,⟨L⟩)
| ICYL[⟨i⟩]         (A,⟨D ei es⟩,⟨L⟩)
| ICONE[⟨i⟩]        (C,⟨D1⟩,⟨D2⟩,⟨L⟩)
| ICONE[⟨i⟩]        (A,⟨D1 ei es⟩,⟨D2 ei es⟩,⟨L⟩)
⟨端面表达式集⟩::= ⟨端面表达式⟩|(⟨端面表达式⟩⟨端面表达式集⟩)
⟨端面表达式⟩::= ⟨⟨NUM⟩⟨端面相关外部关键面名⟩[⟨端面相关内部关键面名⟩]⟨端面特征⟩)
⟨NUM⟩::= 1|2
⟨端面相关外部关键面名⟩::= ⟨外部关键面名⟩
⟨端面相关内部关键面名⟩::= ⟨内部关键面名⟩
⟨端面特征⟩::= CIRCULAR⟨D⟩          /* 实心圆面 */
| RING⟨D1⟩⟨D2⟩          /* 环面 */
| RPCPL⟨D1⟩⟨D2⟩        /* 外部为正多边形,内部为圆 */
| TOLIP⟨D⟩⟨N⟩          /* 齿形面 */

```

图解

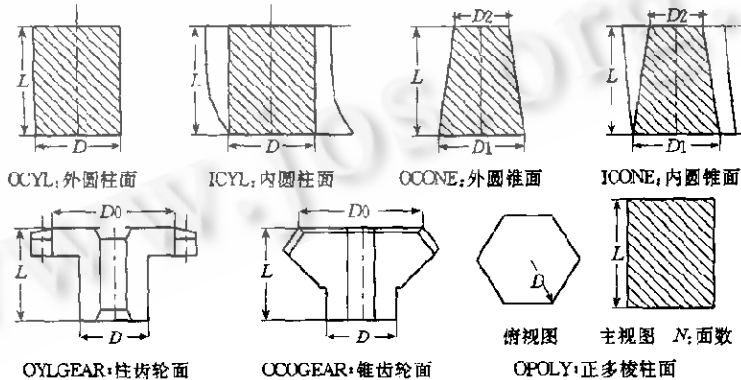


图2 关键面形状、尺寸、符号说明



图3 端面形状、尺寸说明

采用这种知识库/数据库加规则搜索的技术,存在两大优势,①算法复杂度低,无论是从空间还是从时间上来说,复杂度都较低,在空间上只需要存储零件的基本特征,在时间上的复杂度是 $n * m$ (其中 $n$ 为零件数目, $m$ 为每个零件的基本特征数).在实际装配中, $m$ 大多是常数数量级,即 $m \leq C$ .②算法的精确度高,不存在由于计算上的误差而导致的错误发生.

当然,这种算法也有缺点,它只能用于处理一定领域的问题.由于利用知识这种方法的局限性,当然也需要有一种计算零件之间关系的一般算法.由于计算的是两个零件之间的约束关系,因此,最直接的方法就是利用零件的几何信息进行计算,得到它们是否碰撞的信息.这种方法可以适用于任意一个装配件,可以不受经验知识的限制.

### 3 根据装配件的几何信息的算法

在装配的最开始,是接受 CAD 几何造型系统生成的装配件的几何信息.我们知道,这些几何信息很难为我们的规划程序所利用,因为不仅它的信息量庞大,而且它所包含的都是关于点、线、面的一些基本几何信息,其间没有直接明显的联系,不太适用于规划,因此,把这些几何信息转换成规划程序可利用的信息就成为装配系统中不可缺少的一个重要环节.

我们知道,在 CAD 中常用的表示方法是 CSG, BREP 等数据表示,利用这些数据,可以生成我们规划所需要的前趋约束关系,但是存在着精确性和复杂性的问题. BREP 就是边界面的信息,对于装配件来说,表面都是光滑的,但是在 CAD 表示中,大多是采用多边形来近似表示其边界,不可避免地存在误差,因此,在生成前趋约束的时候也可能出现一些错误.用几何信息来生成前趋约束关系还存在着复杂性的问题,无论是时间复杂度还是空间复杂度都是比较大的.由于每个零件的边界信息是很庞大的,因此需要花费较大的空间来存储,同时在计算两个零件的碰撞时,也需要对每个边界进行计算,以得到碰撞关系.在时间上,复杂度的估计是 $N * N1 * N2$ ,其中 $N1, N2$ 为两个零件的边界面个数, $N$ 是零件数目.不难看出,如果单纯地依靠几何信息来得到前趋约束图,基本上是不太可能的.

我们对利用装配件的几何信息生成前趋约束关系作了一个简要的介绍,下面我们给出其具体算法.

#### 算法 1. 计算两个零件的阻碍关系

假设 $a$ 和 $b$ 都是采用 B-rep 表示的多面体(对非多面体零件可取近似), $\tau$ 为三维空间单位向量,判断 $a \stackrel{\tau}{\leftarrow} b$ 问题可以转换成求多面体在平面上的投影和判断平面上多边形是否相交的问题.求解这两个问题的算法都是经典的.<sup>[4]</sup>

#### 阻碍关系判定算法 Blocking( $a, b, \tau$ )

输入: $F_a, F_b, F_c$ 及 $a$ 和 $b$ 的 B-rep 表示, $\tau$

输出: True, 如果 $a \stackrel{\tau}{\leftarrow} b$ , 否则, False.

#### (Step1) 坐标变换 $T$

对 $F_c$ 进行坐标变换 $T$ ,使得 $\tau$ 在新坐标系 $F_c^T$ 中 $\tau^T = [1, 0, 0]$ ;

取 $b$ 上某个顶点作为新坐标系 $F_c^T$ 的原点,使得 $b$ 位于 $F_c^T$ 中 $OYZ$ 平面的右侧,即所有点的 $X$ 坐标均非负.

#### (Step2) 用 $F_c^T$ 中的 $OYZ$ 平面分割 $a$

如果在 $F_c^T$ 中 $a$ 与 $OYZ$ 平面不相交且于其左侧(所有点 $X$ 坐标均非正),则返回 False, goto (Step5);

如果在 $F_c^T$ 中 $a$ 与 $OYZ$ 平面不相交且位于其右侧(所有点 $X$ 坐标均非负),则把 $a$ 标记为 $a^+$ ;

如果在 $F_c^T$ 中 $a$ 与 $OYZ$ 平面相交,则把 $a$ 中位于 $OYZ$ 平面右侧的部分记为 $a^+$ .

#### (Step3) 求 $OYZ$ 平面上的投影

在 $F_c^T$ 中,求 $b$ 和 $a^+$ 在 $OYZ$ 平面上的投影,分别记为 $p_b$ 和 $p_a$ (其中 $p_b$ 和 $p_a$ 为多边形).

#### (Step4) 多边形求交<sup>[6]</sup>

求多边形 $p_b$ 和 $p_a$ 的交,记为 $\Pi = p_b \cap p_a$ .

如果 $\Pi$ 为空或点或线,则返回 False;

否则, $\Pi$ 仍为一多边形,返回 True.

#### (Step5) 结束.

#### 算法 2. 从 BREP 转换成前趋约束图(LCD)

输入: 装配件的 BREP 数据

输出: 装配件的 LCD 图

void brep-cd(BREP brep1, LCD lcd)

(1) 对所有的装配件中的每个零件 part 1, 做:

(1.1) 得到与它不同的一个零件 part 2

- (1.2) 判断在 LCD 图中是否存在 part 1 到 part 2 的边
- (1.3) 若不存在, 则
- (1.4) 得到零件 1 和零件 2 的 BREP 数据 brep1 和 brep2
- (1.5) 计算他们是否在装配路径上相交  
Block(brep1, brep2, path)
- (1.6) 如果相交, 则在 path 的 LCD 图上加上一条 part1 到 part2 的边
- (1.7) 用 warshell 算法扩充前趋约束关系
- (1.8) goto (1.1)

(2) 结束

**算法 3.** 计算两个零件在某个装配路径上是否相交 Block(brep1, brep2, path)

输入: 两个零件的 BREP 信息

输出: 它们交还是不交

BOOL Block(brep1, brep2, path)

- (1) 把 brep1 和 brep2 沿着 path 方向投影, 得到在二维面上的多边形
- (2) 对这些多边形求交.
- (3) 如果交点为空, 或者是点和线,  
则: 返回不交  
否则: 返回交
- (4) 结束

综上所述, 利用装配件的几何信息来计算它们的前趋约束关系既存在优越性, 也不可避免地存在缺陷. 其灵活性决定了它可以应付任何情况, 无论对于什么新物体, 它都可以求出前趋约束图, 而不象第 1 种方法那样, 只能处理已知的特殊情况. 但是它的不精确性和高复杂性也影响了它在实际系统中的应用. 下面举一个不精确性导致错误的例子.

例如“棒子插孔”, 如图 4 所示. 图 4(a)为棒子的 BREP. 图 4(b)为孔的 BREP. 由于 CAD 表示 BREP 的时候总要有-定的精度问题, 所以, 如果精度不够, 就会发生原先不该碰的物体现在计算出来相碰了的情况, 如图 4(c)所示.

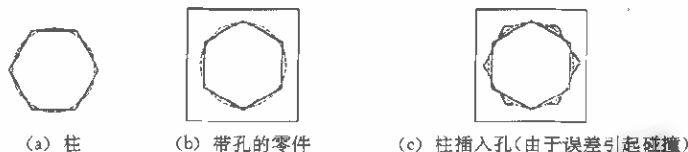


图4 柱插孔实验

此外, 在使用边界面信息计算零件间的阻碍关系时, 还可以采用几何信息中的“三角模型面片”数据来处理. 所谓“三角模型”就是采用三角形, 把零件的表面信息进行划分表示. 目前, 在许多流行的几何造型工具中都提供了“三角模型”数据输出, 计算三角形相交的算法相对来说是十分简单的, 因此, 使用“三角模型”也不失为一种可行的解决方案.

### 4 两种方法的结合

综上所述, 不难看出, 无论是哪种方法都存在自己的优越性, 当然也不可避免地存在着局限性. 因此, 只有把两者结合在一起, 才能取长补短, 达到我们的目的. 大多情况下, 可以采用的方法有以下两种.

#### 4.1 简单结合

以知识库/数据库的规则搜索方法为主. 对于一个新的装配件, 主要还是要先考虑其特征, 根据它进行计算, 求出已知的约束关系来. 因为根据统计装配体中的约束关系在上面提到的约束关系中所占的比例还是很大的. 遇到不在我们的知识库中的特殊的装配件, 就交给几何方法来处理, 同时对通过几何方法解决的方案进行归纳总结, 总结出的规则存入知识库来指导今后的前趋约束图的生成. 这样, 既提高了速度, 也解决了局限性的问题. 这种方法主要是以知识规则的方法为主. 简单结合分为以下两种: (1) 以知识为主(如图 5 所示); (2) 以几何方法为主(如图 6 所示).

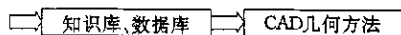


图5 以知识为主的算法

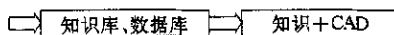


图6 以几何方法为主的算法

### 4.2 紧密结合

首先利用知识库、数据库规则搜索方法对装配件进行计算,遇到特殊装配件后,根据在数据库中零件的 CSG 信息,把装配零件分解成一些基本部件的几何操作(并、交、差),然后利用几何求交来计算各个基本部件之间的碰撞关系。这种方法可以避免前一种方法中单纯利用几何方法计算引起的复杂性问题,因为计算基本部件的碰撞的复杂度是较低的,而且基本组件的 BREP 数据也比较容易得到。在这个方法中主要是以几何方法为主。

在这个系统中,采用了两种方法相结合的策略,各取所长,既充分利用了知识的优越性,又充分结合了几何算法的灵活性,使得 LCD 的生成既快又灵活。可见,在实际系统中,单纯的计算有时并不都占主导地位,适当地加入相应的知识作为指导,再结合现有的一些数学分析工具,才能达到最佳的效果。缺乏知识指导的工具必然会缺乏实用性,脱离工具,单纯利用知识又会导致其通用性较差。如何合理地将两者结合起来,成为系统集成的关键。

## 5 实 验

从 1987 年开始,清华大学计算机系人工智能实验室就开始进行机器人自动装配系统的研究,这些项目属于国家 863-512 智能机器人主题,系统的研究分为 3 个阶段。

### (1) 理论研究阶段

- ① 机器人任务规划方法研究(1987~1991 年)
- ② 机器人任务、路径规划方法研究(1991~1992 年)

### (2) 离线规划阶段

- ③ 基于知识的机器人装配规划系统(1993~1994 年)
- ④ 交互式机器人装配规划系统(1994~1996 年)

### (3) 在线规划阶段

- ⑤ 任意时间算法(1996~1998 年)

整个系统的框图如图 7 所示。系统通过人机交互接口把人和机器规划系统联系成一个有机整体,从 CAD 图纸开始,到装配序列的生成,送给规划调度,调度的结果送给仿真系统以及实时监控,通过反馈,及时修改原有规划中不合理部分,同时在实际运行环境中遇到意外情况时,调用任意时间算法,尽快求出较为合理的解决方案。

在整个系统集成方面,自动化程度占有十分重要的地位,整个系统实现的目的就是为了让机器人能够在实际环境中自动进行机械装配。当然,不能否认实际装配中存在很大难度,其中包括时间、环境等复杂因素,因此,完全自动化的可行性不大,所以,在适当的环节引入人工干预也是必不可少的。在系统实际设计中,我们发现在整个系统的最前端(装配前趋有向图的生成)存在较大的难度,它受到时间因素的限制,完全采用几何计算是不太可行的。因此,我们引入了专家的经验知识,设计了一种两类方法相结合的算法,同时,对一些计算时间作了初步统计。

对于以上提出的方法,相应建立了 3 类程序:一种是单纯几何计算方法,一种是基于知识库/数据库的算法,第 3 种是两者的结合。

在几何算法中,出于一般性讨论,系统在测试过程中采用随机生成三角模型面数据的方法。表 1 是实验的一些结果。

表 1 计算两个用三角模型表示的零件沿着某个装配路径是否碰撞的算法的时间估算(纯几何算法)\*

零件中三角面片个数	100	300	1 000
算法计算时间(s)	0.824 175	7.197 802	80.549 451

同时,我们通过几何算法对采用三角模型表示的装配件计算其装配前趋约束图,得到一些实验数据,见表 2。

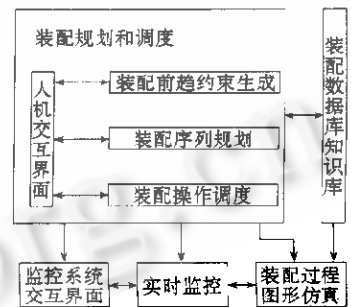


图7 机器人自动装配系统框图

\* 时间统计在 Pentium 133,32M RAM DOS 环境下测试得到。

表 2 计算用三角模型表示的装配前趋约束图算法的时间估算(纯几何算法)\*

装配件中零件数目	5	10	20
装配路径个数	2	5	10
计算 LCD 时间(s)	165	18 220	153 000**

(假设每个零件用 1 000 个三角型来模拟其外表面)

通过以上的实验数据,我们可以看出,单纯地使用几何知识来计算装配件的前趋约束图基本上是不太可行的,因此,在实际应用中大多还是采用知识库/数据库的解决方案,它的算法复杂度很低,也就是  $O(N)$ ,其中  $N$  为特征数.

由于在实际计算中,特殊零件或者特殊的特征是比较少的,大多数装配件由一些系统定义的基本零件组成,不妨假设在实际装配中存在 5% 的特殊零件,因此得到一些实验数据,见表 3.

表 3 采用几何知识和数据库/知识库相结合的方法计算装配前趋约束图算法的时间估算

装配件中零件数目	5	10	20
装配路径个数	2	5	10
计算 LCD 时间(s)	8	900	7 600

(假设每个零件用 1 000 个三角形来模拟其外表面)

以上数据均是在假设系统需对所有的三角面片计算其相交性之后才能得出零件是否碰撞的结论,而在实际应用中,如果两个零件发生碰撞,那么在计算过程中就会发现某些三角形相交,因此,实际计算时间要远小于上面我们给出的估算时间.同时,如果对监测碰撞过程采用从较粗粒度到较细粒度的监测,那么速度也将大大加快.

### 6 总 结

总之,求装配件的前趋约束是为了让装配过程能够实现自动化.适应现在工业生产的需要.利用经验知识结合装配件的几何信息来处理装配件之间的约束关系,是一种可行的方法.在装配序列自动生成方面,国内外都进行了许多方面的工作,但是在从 CAD 图纸自动生成前趋约束图的实现上都存在一定的问题,不是在速度上,就是在算法通用性上存在或多或少的问题.基于这些问题,我们实现的结合算法解决了一部分问题,提高了自动装配的效率.

同时,通过前趋约束图生成的方法,我们不难看出,知识在实际算法中处于十分重要的地位.离开知识的算法显得比较盲目,缺乏算法指导的知识适用面不广.如何合理地将两者结合起来,成为解决工程问题的关键.

### 参 考 文 献

- 1 鲁涛.智能机器人装配系统——体系结构、任务规划和原型系统[博士论文].清华大学,1993  
(Lu Tao. Intelligent assembly robot system——architecture, task planning and prototype system [Ph. D. Thesis]. Tsinghua University, June 1993)
- 2 De Fazio T L, Whitney D E. Simplified generation of all mechanical assembly sequences. IEEE Joint of Robotics and Automation, 1987, RA-3(6):640~658, RA-4(6):705~708
- 3 Baldwin D F, Abell T E, Lui M-C *et al.* An integrated computer aid for generating and evaluating assembly sequences for mechanical products. IEEE Transactions on Robotics and Automation, 1991,7(1):78~94
- 4 Huang Y F, C S G Lee. A framework of knowledge-based assembly planning. In: Proceedings of IEEE International Conference on Robotics and Automation. 1991. 599~604
- 5 Delchambre A, Wafflard A. An automatic, systematic, systematic and user-friendly computer-aided planner for robotized assembly. In: Proceedings of IEEE International Conference on Robotics and Automation. 1991. 298~592
- 6 孙家广,陈玉健,辜凯宁.计算机辅助几何造型技术.北京:清华大学出版社,1990  
(Sun Ja-guang, Chen Yu-jian, Gu Kai-ning. Computer Aided Geometric Modeling. Beijing: Tsinghua University Press, 1990)

\* 时间统计在 Pentium 133,32M RAM DOS 环境下测试得到.

\*\* 此数据是估算得到的.



## The Generation of Assembly Precedence Constraint Based on Knowledge

CHEN Zheng ZHANG Bo

*(Department of Computer Science and Technology Tsinghua University Beijing 100084)*

*(Laboratory of AI Technology and Systems Tsinghua University Beijing 100084)*

**Abstract** The generation of assembly precedence constraint is the key step of assembly planning and the most difficult one as well. In this paper, the implementations of some methods of generation of assembly precedence constraint are presented at first. The advantages and disadvantages of these methods are discussed as well. Then, one method which combines the knowledge of geometry and assembly experience is presented. By using the method, the assembly knowledge base and database are established. Several feasible solutions are provided at the same time. Finally, a conclusion is reached. The advantages of the method and suggestions on further research are provided as well.

**Key words** Automatically assemble, blocking unit, precedence constraint, geometry knowledge, triangle model, assembly knowledge bases.