

文章编号: 1009-038X(1999)04-0095-04

三维约束装配技术

朱文华¹, 过小容²

(1. 无锡轻工大学机械工程系, 江苏无锡 214036; 2. 无锡联发信息产业, 江苏无锡 214001)

摘要: 基于微机三维计算机辅助设计 (CAD) 系统, 针对三维约束装配的过程, 阐述了约束装配过程中零件引用、引用顺序、施加装配约束、装配分析、装配编辑修改、二维装配图生成的主要技术。

关键词: 三维; 计算机辅助设计; 约束; 装配

中图分类号: TP391.72 **文献标识码:** B

随着科技的发展, CAD技术日益被重视, 在微机上已从二维向三维发展, MDT (Mechanical Desk Top) 是美国 Autodesk 公司推出的融合二维和三维设计并带有装配功能的机械设计平台, 它是集 AutoCAD 与参数化实体造型、曲面造型、装配造型、二维和三维双向关联绘图以及 IGES 转换器等模块为一体的机械设计系统。其中装配造型是极为重要的模块, 本文就此对装配的建模过程及约束装配的主要技术进行研讨^[1-3]。

1 约束装配的过程

装配是三维参数化 CAD 系统的重要技术。MDT 装配是带约束装配, 它不同于积木式装配, 装配中的零部件通过点、边、轴或面的关联定位确定它们之间的相对位置和配合关系。MDT 的装配对象是零件或部件, 零件的模型通过参数化的实体造型生成, 装配时零部件先引用到装配图, 按照自下而上设计的方法排序, 然后通过施加装配约束消除自由度进行装配, 装配后进行装配分析, 装配后的零部件可以编辑修改, 编辑后仍然保持装配关系不变, 最后生成二维装配图。

MDT 约束装配的过程如图 1 所示。



图 1 MDT 约束装配的过程图

收稿日期: 1999-03-04; 修订日期: 1999-09-25

作者简介: 朱文华 (1968年 4月生), 男, 江苏无锡人, 博士研究生, 工程师。

2 零件的引用

零件的引用类似于在 AutoCAD 中,使用插入命令插入一个块,所有该零部件的信息随着零件定义记录在部件的装配模型中.插入到装配中的一个零件或部件是实际零件模型的一个参考备份,在一个装配里零件可以多次引用,引用的零件可以是当地零件(存在当地图形文件中),也可以是外部参照零件(存在外部图形文件中),零件修改时,装配件自动更新.第一个引用的零件称为基础零件.

零件引用有 3 个方面的重要作用,即: 1)提高零件模型的使用效率.某个零件模型完成后,可能会被多个用户使用,在整个设计过程中,相同的零件会在其它一些装配中使用. 2)节省零件模型的磁盘存储空间.如果在装配中需要多次使用一个零件模型,在系统中只需存储一个零件模型,这将大大节省磁盘空间. 3)提高装配模型的编辑修改效率.当修改一个零件定义时,所有对该零件的参考均自动修改,这样可减少人工修改每个引用处的工作量.

3 装配结构及零件引用顺序

装配结构可以用两种方式显示某个装配设计:一个是逻辑结构,一个是图形结构.逻辑结构表达了组成该装配模型的零件、子部件之间的层次关系,就象目录的树型结构.装配模型的逻辑层次结构通过装配树表达.

根部件:根部件是在生成一个新的装配图时自动产生的,是装配的最顶层.

子部件:装配中的一个逻辑组,由零件构成的子装配,子部件可以嵌套任意深度.

零件:层次结构中的最低一级,是单一实体,不能拆成更小的实体.

图形结构是某个装配模型的几何表示,直观地用分解图表示.如图 2 所示的法兰联接装配模型,由法兰、法兰盖、垫片、螺栓、螺母组成.

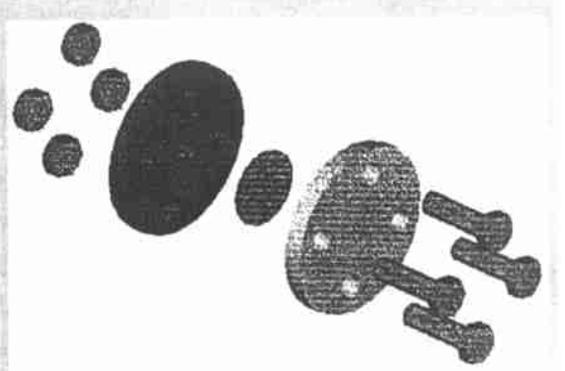


图 2 法兰联接装配模型的图形结构

零件引用顺序就是零件插入到装配模型中的顺序.用浏览器(Browser)可以显示零部件的引用顺序并且还可以对引用的零件重新排序,引用顺序应与实际装配顺序相同.

装配树中零件的引用顺序直接影响装配模型约束的解释,MDT的装配约束使用一个参数算法器,该参数算法器按零件引用顺序解算约束.基础零件是装配树中位于根部下的最顶端零件.基础零件的自由度为零,是静止不动的.基础零件的作用像一块“磁铁”,“吸引力”是装配约束,按装配树自上而下顺序把其他引用零件“吸引”过来,也可以对两个或两个以上与基础没有约束关系的零件施加约束,它们组成一个浮动集.

4 施加装配约束和装配分析

MDT创建的是参数化装配,可以在一个零件上多次施加约束,对零件施加约束就是减少零件自由度或运动类型,限制零件之间的相对运动,并且通过点、边、轴或面的关联定位确

定装配模型中零件的位置和配合方式。

MDT提供 4类装配约束:配合 (Mate)、表面平齐 (Flush)、对准角度 (Angle)、插入 (Insert)。配合约束使一个零件的平面或轴与另一个零件的平面或轴在一个指定的方向上相一致,它减少一个移动自由度;表面平齐约束使两个平面共面并在同一方向上对齐;对准角度约束使一个零件的平面或轴与另一个零件的平面或轴在一个指定的方向上成一定的角度;插入约束在指定的方向上对齐两个圆的圆心使两个平面配合,去除移动自由度。对于零件,每增加一个约束便减少一些自由度,自由度符号反映了零件的引用顺序和零件如何运动。在装配中施加何种约束取决于具体的设计目标和设计者的经验。图 3所示的是法兰联接施加装配约束后的装配模型。



MDT对装配的分析包括干涉分析和质量特性分析。MDT可以对装配模型中的零部件进行干涉分析,如果存在干涉,系统显示干涉区域和相互干涉零部件的名称。根据零部件的干涉分析结果,确定是否修改编辑现有的装配模型。在概念设计阶段,质量特性计算可提供关键性设计所需的信息,MDT可以为选中并指定材料的零件计算质量特性,质量特性计算对于产品设计是必不可少的。

5 装配的编辑修改

设计或规格的改变要求装配进行相应的编辑修改,装配的编辑修改就是编辑组成该装配模型的零件或编辑装配约束。组成装配模型的零件包括当地零件和外部参照零件。当地零件可以外部化,外部参照零件也可以当地化。

5.1 编辑当地零件

如果要修改一个零件,将该零件当地化,然后在当前图形文件中编辑该零件,编辑完成后更新零件。在当地图形文件所做的修改,仅对当地零件起作用,不影响它的外部图形文件。

5.2 编辑外部参照零件

MDT的零件模型可以在多个装配设计中以外部参照的方式被引用,外部参照零件的编辑可以当地化后编辑,也可以打开被编辑零件的外部图形文件,编辑该零件模型的特征,零件模型修改后存盘,然后再打开装配模型的图形文件,所有引用该零件的地方自动更新。零件编辑修改后约束关系保持不变。

5.3 编辑装配约束

编辑装配约束包括修改、增加、删除约束,编辑装配约束可以重新对齐零件或修改装配关系,在编辑装配约束时,可使用 Move 命令把零件移开使零件清晰可选,编辑约束后可用自由度检查和干涉检查约束的合理性。

6 装配图的生成

为了清晰地表达部件中各零部件的装配关系,在生成二维视图之前,可以先生成部件的装配场景,场景是一个分解视图,它把部件或子部件的零件分离,以显示它们是如何装配的以及它们的逻辑关系。

MDT在图纸空间中可以自动生成二维视图,其中包括正交视图,轴侧图,半剖视图,全

剖视图,阶梯剖,旋转剖和局部放大图;按照零件装配顺序,MDT自动标注零件序号,并自动生成零件明细表。

图幅和标题用 MVSETUP命令设置,也可将过去 AutoCAD上做的图幅和标题栏用插入命令装入到 MDT环境中供用户调用。

7 结语

微机 CAD系统从二维向三维发展是必然趋势,本文基于微机三维 CAD系统,从三维约束装配的过程出发,阐述了约束装配过程中零件引用、引用顺序、施加装配约束、装配分析、装配编辑修改、二维装配图生成的主要技术。此文对 CAD技术的应用和三维 CAD技术的普及推广有推动作用。

参考文献:

- [1] America Autodesk Inc. Technical specifications assembly modeling [EB/OL]. <http://www.autodesk.com/products/desktop/techspec.htm#assembly>, 1999-01-25.
- [2] 朱文华.轻工机械 CAD[M].北京:中国轻工业出版社,1998:176~198.
- [3] 陈伯雄,张苏苹. AutoCAD R14中文版高级应用教程 [M].北京:电子工业出版社,1999.

Three Dimension Constrain Assembly Technology

ZHU Wen-hua¹, GUO Xiao-rong²

(1. Department of Mechanical Engineering, Wuxi University of Light Industry, Wuxi 214036 2. Wuxi Aligrow Information, Wuxi 214001)

Abstract Basing on three dimensional CAD system on PC, this paper illuminated the main technology in the course of constrain assembly, that includes instancing part, instancing order, applying assembly constraints, analysing assembly editing assembly, creating assembly drawing views.

Key words three dimension; CAD, constrain; assembly