

## 基于特征重排的协同设计系统研究

聂海丽<sup>1</sup>, 王晓国<sup>1</sup>, 王敏<sup>2</sup>

(同济大学 计算机科学与技术系, 上海 200092;

2. 中国石油管道公司沈阳输油分公司, 辽宁 沈阳 110031)

**摘要:**随着协同虚拟环境概念的提出,使得多用户之间基于特征的建模设计成为可能并迫切需要一个有效、灵活并支持异地分布的多用户协同工作的系统。在文献[2]提出的特征重排思想的基础上对协同设计系统进行研究,提出了一个基于特征重排的协同设计应用系统模型,详细探讨了模型系统的结构,重点阐述了该系统模型中当用户不按照初始设计顺序提交特征造型信息时,服务器如何依据特征重排的思想对特征进行布尔操作保持与原始造型一致,这对于提高协同设计的效率有重要意义。

**关键词:**特征;协同设计;有效特征量;令牌;分布式结构

**中图分类号:**TP391

**文献标识码:**A

**文章编号:**1673-629X(2007)11-0019-04

## Research of Feature - Rearranged Collaborative Design System

NIE Hai-li<sup>1</sup>, WANG Xiao-guo<sup>1</sup>, WANG Min<sup>2</sup>

(1. Computer Science &amp; Technology Department of Tongji University, Shanghai 200092, China;

2. Petro China Pipeline Shenyang Sub - Company, Shenyang 110031, China)

**Abstract:** The emergence of concept of CVE makes collaborative feature modeling among multi - users possible and arises the urgent needs of developing system to support collaborative work of multi - users distributed at different places efficiently and vividly. Presents a model of collaborative design system according to the method of feature - rearrangement given in reference[2]. Architecture of the model is discussed in detail, and the approach of how to make the result model be consistent with the required model after the Boolean operation on server when the order of the submitted features are different from the original design order is introduced. This model is important for improving the efficiency of collaborative design.

**Key words:** feature; collaborative design; effective feature volume; token; distributed structure

## 0 引言

在基于特征的造型方法中,特征是基本的单元<sup>[1]</sup>。一个对象的造型过程就是在基本特征的基础上不断添加新的特征,最终得到所需要的造型。比如以立方体为基本特征,在此基础上做布尔加运算添加一个圆柱体1,然后用一个新的圆柱体2和原先生成的造型做布尔减,实现打洞操作来得到最终的造型,过程如图1所示。协同设计系统问世以来,多个造型任务可以分配给不同的客户端来完成,在这样的协同设计过程中,圆柱体1和圆柱体2可以由两个不同的客户端来完成,然后再传到服务器通过布尔运算得到最终造型。

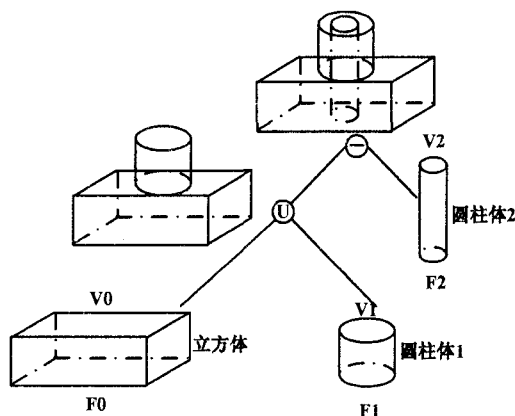


图1 造型过程举例

现在假设圆柱体1和圆柱体2分别由 client1 和 client2 完成,并且 client1 比 client2 先完成造型并提交给服务器做布尔操作,这时就会得到和图1一样的效果。但是,如果 client2 比 client1 先完成造型并提交给服务器,就会出现图2的情况,这与设计的初衷是不符的。

收稿日期:2007-01-26

**作者简介:**聂海丽(1981-),女,河北石家庄人,硕士研究生,主要从事企业信息化、图形图像处理方面的研究和应用;王晓国,博士,副教授,硕士研究生导师,主要从事企业信息化、图形图像处理方面的研究和应用。

为了解决这个问题,虽然可以再生成一个圆柱体,并通过布尔减来修补以达到预期的效果,但是由于现实中的复杂性,总是这样修改,不仅工作量大而且其正确性也很难保证。为了使实际应用中协同设计的结果符合设计的初衷,在特征重排思想的基础上提出了一种能够实现无序造型的协同设计系统模型,这个系统模型对在保证造型结果一致性的前提下提高 CAD 协同设计的效率和灵活性有重要意义。

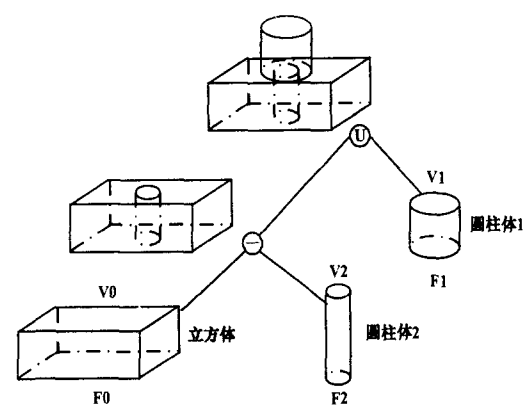


图 2 Client2 比 Client1 先提交

### 1 特征重排思想概述

任意重排特征的顺序可以分解为一系列的特征重定位,而且每一次重定位可以被进一步分解为一系列相邻特征的交换。所以,要解决特征的重排问题,关键是要找到一种方法使得交换两个特征后仍能得到相同的布尔运算结果。有效特征量便成了这个思想的核心。有效特征量是部分或全部的特征量,为的是在交换特征的顺序后仍能得到相同的布尔运算结果。当按照特征造型的原始顺序进行造型时,每个特征的有效特征量就是全部的特征量。然而,当重排特征时,一个特征的有效特征量可能会减小为原来特征量的一部分,从而使最终的造型效果符合设计的初衷<sup>[2]</sup>。

在基于特征的造型中,形状特征被划分为加的特征量和减的特征量两种类型。用“+”来表示加操作,用“-”来表示减操作。如果用  $\otimes_i$  表示 + 或 - 的布尔操作,用  $V_i$  表示特征  $F_i$  的原有特征量,那么,交换两个布尔操作的结果可以由下面的公式(1)表示:

$$V_0 \otimes_1 V_1 \otimes_2 V_2 = V_0 \otimes_2 V'_2 \otimes_1 V'_1 \quad (1)$$

这里  $V'_1$  和  $V'_2$  是  $V_1$  和  $V_2$  的有效特征量。这个结果也可以用公式(2)表示为:

$$V_0 \otimes_1 V_1 \otimes_2 V_2 = V_0 \otimes_2 V_2 \otimes_1 (V_1 - \varphi(\otimes_1, \otimes_2) V_2) \quad (2)$$

这里:

$$\varphi(a, b) = \begin{cases} 1 & a \neq b \\ 0 & a = b \end{cases}$$

用  $M_n$  表示对  $n+1$  个实体模型应用  $n$  次布尔操作后得到公式(3) 所示的最终模型:

$$M_n = \prod_{i=0}^n \otimes_i V_i \quad (3)$$

这里:  $\otimes_0 V_0 = \emptyset \otimes_0 V_0$

如果相应于特征  $F_j$  的操作  $\otimes_j V_j$  被移到第  $m$  的位置上,  $M_n$  可以用下面的公式(4) 表达:

$$M_n = \left( \prod_{i=0, i \neq j}^m \otimes_i V_i \right) (\otimes_j (V_j - \sum_{l=1}^{m-j} \varphi(\otimes_j, \otimes_{j+l}) V_{j+l})) \left( \prod_{i=m+1}^n \otimes_i V_i \right) \quad (4)$$

参考文献[2] 中已经对公式(1) ~ (4) 进行了完整证明,这里不再叙述。

### 2 一个特征重排协同设计应用系统模型

随着 Internet 的迅速发展,在 CAD 模型的开发过程中,伴随着越来越多的异地协同开发情况的出现。基于 Web 的计算机网络技术的发展,提供了一种能实现群组分布的 CAD 设计专家异地协同求解复杂问题,合作完成产品设计的物质基础。而在网络环境下计算机支持的合作工作体现了信息时代人们工作方式的群体性、交互性、分布性和协调性的客观需求<sup>[3,4]</sup>。因而,分布式协同设计研究已成为国内外计算机工程应用研究中最活跃的前沿方向之一,而协同设计环境的研究则被认为是当前 CAD/CAM 技术发展的一个重要趋势。笔者参与在基于特征的造型技术中,应用了特征重排的思想,提出了一个应用特征重排技术的协同设计系统模型。

#### 2.1 系统模型结构

##### 2.1.1 系统模型结构描述

文中所探讨的系统模型是一个典型的 C/S 模型。图 3 是所提出的特征重排协同设计应用系统模型,该系统能够支持网络中多个客户端连接到服务器。在服务器端布置了协同管理器和模型管理器。其中,协同管理器包含一个部件模型,而模型管理器则包含了一个会话模型,这两个模型互相配合以协助协同设计功能的实现和完善。

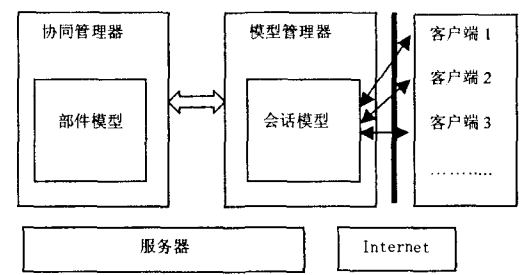


图 3 特征重排协同设计系统模型示意  
系统模型中每一个设计对象,即部件,对应着唯一

的一个部件模型,它位于协同管理器中,用于存储全局的特征模型。所有从客户端提交的特征模型信息都将在协同管理器中依次完成其对应的操作。例如当协同管理器得到一个客户端传来的特征信息后,它会将这个特征信息所代表的特征模型与当前的全局特征模型做相应的布尔操作,从而得到新的全局特征模型。而模型管理器则记录着一个会话模型,会话模型实际上是协同管理器中部件模型的一个快照。这个模型一方面可以维护客户端的特征信息并且协调各个客户端的工作,另一方面可以实时地更新全局特征的造型。会话模型从服务器传向客户端,每个客户端都可以查看此模型并且可以抓取一些特征信息。

### 2.1.2 系统工作流程

在系统工作之前,任务已经被分解为一系列任务模块,即特征,并且对每个特征按原始顺序进行按序编号,这个编号与模块都是公有变量,每个客户端都可以访问。客户端在设计好模型之后为其打上编号,并将其与操作信息一起发送到服务器由模型管理器的会话模型进行冲突检测与控制以及特征重排。确保无冲突之后再交给协同管理器进行操作,更新协同管理器中的部件模型。

客户端与服务器的交互由模型管理器中的会话模型来完成。客户端向服务器发送请求,会话模型判断是查看操作还是有新的特征到来,若是查看操作,会话模型从协同管理器中取出全局特征(部件模型)并发送客户端。若有新特征到来则将收到的特征及操作传送给协同管理器,然后协同管理器按照特征重定位的思想进行造型。

将全局特征量与新特征量进行布尔操作之后,会话模型及时将所得的结果广播给所有客户端以实现造型信息的一致性。流程如图 4 所示。

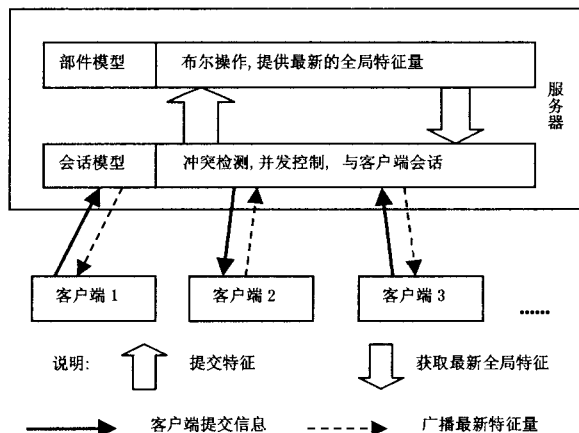


图 4 流程示意图

### 2.1.3 客户端的状态

在系统的整个工作流程中,客户端只处在两种状

态:操作状态和会话状态。

如果客户端已经完成了本地部件的设计,此时需要将此部件发送到服务器进行合成操作。而操作状态是指客户端已经获得冲突控制机制的授权,他所提交的操作正在被服务器执行的状态。

会话状态是相对于操作状态而言的。此时客户端正在从会话模型接收造型的相关信息或者查看全局的部件模型的状态。系统中造型信息的一致性主要是靠客户端与服务器端的会话来实现的,因而会话状态是至关重要的。

### 2.2 信息传递

在协同设计人员进行同步协同设计时,为了保证协作人员之间的信息交流清晰明了,在传输信息时除了传输零件的各种特征信息和特征约束信息外,还需传输对特征的操作信息<sup>[5]</sup>。在本模型中规定协同设计人员能够进行的特征操作主要是特征布尔加和布尔减。

由于是多个客户端进行协同设计,在信息提交到服务器时会出现互相冲突的情况,这就需要表明事件的先后关系,所以传输的信息中还需要打上时间戳以提高冲突控制机制的效率。

### 2.3 冲突控制机制

在同步协同设计过程中,由于服务器同一时间只能处理一次操作,多个协同设计人员对同一特征进行的操作可能互相冲突,为了避免由于相互冲突引起的不一致,需要提供能体现协同设计人员之间交互性的并发控制机制。本系统模型为不同的特征造型预先定义一定的优先级,并且采用令牌控制机制进行冲突控制。令牌是一种特殊标记,只有令牌持有者才能进行特征操作。

开始协同设计时,由会话模型持有令牌,其他协同设计人员如须拥有令牌,可向会话模型提出申请,在会话模型同意其申请后,就将令牌转交给提出申请的协同设计人员。当有多个协同设计人员提出申请时,将按预先定义的优先顺序分配令牌;在同一优先级别中,按申请时间的先后顺序,先申请的协同设计人员先得到令牌。协同设计人员进行特征操作时,以知识查询与处理语言(Knowledge Query Manipulation Language, KQML)格式将所有信息提交给系统模型管理器。模型按一定算法对信息作适当处理后,将操作信息向所有在线参与协同工作的设计人员以广播的方式传输出去;同时,将所有操作信息及更新后的数据作为新版本存入共享数据库,并向其他离线的相关人员发出数据更新信息,通知他们适时更新数据。参与协同工作的设计人员收到信息后立即更新其本地数据库。

## 2.4 特征重定位机制

在协同设计的过程中,采用的是分布式结构,设计人员分布在不同的客户端,而且各自都是完成一些局部的特征造型,然后再提交给服务器进行进一步的集成操作。按照特征重排的原理设计出这个系统,保证了客户端在提交特征量时无需按照原始顺序提交,从而大大提高了系统的灵活性。

对于每一次特征操作,客户端的交互过程如下:一个设计者(即客户端)首先设计出特征模型,然后选取合适的位置和方向约束(即定位特征和方向)。当这些完成后,设计者向服务器提交特征信息,并获得显示结果。该显示结果是由服务器进行相应的布尔操作后传送回客户端的。当有多个客户端时,由于任务被分解,在服务器端接收到的特征顺序就会被打乱。如果服务器还是按照原始操作执行,将会得到一个错误的结果。所以此时的布尔操作需要考虑到特征重排的顺序问题。

当一个特征到达时,首先对其进行特征重排。协同管理器利用每个特征量的唯一编号结合基于特征的重排思想实现设计结果的正确性。当服务器正在处理一个客户端的提交时,它首先提取出此提交特征  $F_i$  的编号,并将其与已经处理的特征编号进行比较。只有那些本来应该在其后处理而现在提前处理了的特征才有可能影响对此特征的处理。因此需要对此特征进行重排,以防止它与这些“反序”(相对于此特征前后顺序发生变化)的特征出现不必要的操作。具体的重排过程是:对所有编号大于它的特征  $F_j, F_i$  的布尔操作改为:

$$\otimes_i V'_i = \otimes_i (V_i - \sum \varphi(\otimes_i, \otimes_j) V_j) \quad (5)$$

$$\text{这里: } \varphi(a, b) = \begin{cases} 1 & a \neq b \\ 0 & a = b \end{cases}$$

就是用这些  $F_j$  对  $F_i$  进行重新修剪然后再与全局特征量进行布尔操作。若不存在这样的  $F_j$ , 则不进行修剪,即  $\otimes_i V'_i = \otimes_i V_i$ 。用  $M_n$  表示对  $n+1$  个实体模型应用  $n$  次布尔操作后得到的最终模型:

$$M_n = \prod_{i=0}^n \otimes_i V'_i \quad (6)$$

经过特征重排之后,特征造型的结果仍然是正确的。采用特征重排的思想可以很大程度地提高协同设计的效率。在本系统模型中,将特征重排与令牌机制结合起来,一方面可以保证整个协同的过程井然有序,另一方面使得服务器无需等待特征的按序到来,可以大大提高协同设计的效率。

## 3 结束语

随着计算机网络技术的快速发展和特征建模技术研究的深入,计算机支持的协同特征建模将是 CAD 技术发展的一个重要方向,也是协同设计理论研究与发展的必然趋势。因为当前多数基于特征的协同设计造型系统不能异步处理多个客户端提交上来的特征造型。文中主要讨论了客户端不必按照原始的特征造型顺序来提交,先完成造型任务的可以先提交,而造型的结果却能保持与原始建模结果完全一样的设计,所提出的这种重排特征顺序的模型思想有助于增加协同设计系统的灵活性,在协同设计中具有重要意义。但是,文中所提出的模型和设计思路还处于理论阶段,还有待进一步实现。

### 参考文献:

- [1] 尚建忠,孟凡磊,唐力,等. 基于三维 CAD 平台的实时协同设计方法[J]. 机械设计, 2005, 22(8): 43-45.
- [2] LEE SANG HUN. Feature-Based Multiresolution Modeling of Solids[J]. ACM Transactions on Graphics, 2005, 24(4): 1417-1441.
- [3] 周自强,李晓光,沈连娟. 基于特征语言的远程协同实体建模研究[J]. 现代制造工程, 2003(2): 24-26.
- [4] 李伟刚,莫蓉,杨海成,等. 基于 Web 的零件特征共享技术研究[J]. 计算机辅助设计与图形学学报, 2002, 14(10): 1-5.
- [5] 汪惠芬,张友良,曹健. 基于特征的协同设计[J]. 计算机辅助设计与图形学学报, 2001, 13(4): 367-372.

(上接第 18 页)

难度。由于协议数据转化主要是软件实现,在软件的实时性能方面,如何定义各总线帧的实时约束和全局优先级,实现优化调度是下一阶段开发过程中进一步研究的重点。

### 参考文献:

- [1] 夏德海. 现场总线现状及其应用[J]. 中国仪器仪表, 1998(3): 2-3.
- [2] 郭宽明. 现场总线应用选编[M]. 北京: 北京航空航天大学出版社, 2004.

学出版社, 2004.

- [3] 丁磊,费敏锐,陈维刚. 现场总线协议转换机理及实现[J]. 自动化仪表, 2005(7): 1-3.
- [4] Labashov M, Pratl G, Sauter T. Applicability of Internet protocols for fieldbus access[C]//Proc. 4th IEEE International Workshop on Factory Communication Systems. Varsalas, Sweden: [s. n.], 2002: 28-30.
- [5] 张万升,刑涛,刘尉悦. 基于 VxWorks 的多 DSP 系统的多任务程序设计[J]. 单片机与嵌入式系统应用, 2001(6): 9-11.