

以网络为中心的语义特征造型协同设计

王远志¹, 王少威², 孙立铸³

(1. 安庆师范学院计算机系, 安庆 246011; 2. 武汉科技大学计算机系, 武汉 310027; 3. 哈尔滨理工大学计算机学院, 哈尔滨 150080)

摘要: 在网络化同步协同设计环境中, 如何实现用户系统之间的实时数据交换、维护特征语义的一致性, 成为语义特征造型协同设计的关键问题。该文介绍了基于细胞元表示的语义特征模型, 根据该造型技术的分层体系结构, 基于对协同环境下特征操作的分析, 采取了传输造型操作命令的策略, 为解决特征语义一致性问题, 提出了一种用操作特征管理器调度操作执行的方法, 并给出了一个实例。

关键词: 细胞元; 语义特征造型; 特征依赖图

Collaborative Design of Semantic Feature Modeling on Network-centric

WANG Yuanzhi¹, WANG Shaowei², SUN Lijuan³

(1. Department of Computer, Anqing Normal College, Anqing 246011; 2. Department of Computer, Wuhan University of Science and Technology, Wuhan 310027; 3. School of Computer, Harbin University of Science and Technology, Harbin 150080)

【Abstract】 Within network collaborative design environment, real-time data exchange and feature semantic conformity become a key problem for collaborative design on semantic feature modeling. A semantic feature modeling is proposed, according to the structure of semantic feature model, the strategy of the modeling operation commands being transmitted is adopted. In order to maintain feature semantic conformity, a new method by which OFM is introduced to control operation's executing is suggested. An example is introduced.

【Key words】 cellular; semantic feature modeling; feature dependent graph

自从 Rafael Bidarra 等 1997 年首次提出语义特征造型技术后, 该技术已经在工程实践中得到了广泛应用。伴随着网络技术的飞速发展和 Internet 的成熟, 面向网络的设计环境作为产品开发生的新典范得到了同行的认可。当前有许多支持协同设计的产品和实验系统。CADDAC 系统中针对多客户协同问题使用了 Model-Controllers-Viewers 设计模式。然而在某一时刻, 只能有一个主客户可以调用服务器的造型功能, 其他的客户只能观察模型。CollIDE、NetFeature、Cooperative、ARCADE、CSCW-Feature M、TOBACO 都是研究性实验系统, 它们给出了处理协同设计问题的一些好方法, 但并发问题仍有待解决。在 CSM 系统中, 每个客户端都保留了产品模型的一个拷贝。当一个客户端修改了模型, 其结果被传送回服务器, 然后再发送到其他客户端。它采用令牌方法以保证同步。OneSpace 是一个已具有一些协同功能的造型系统, 但由于其服务器端的几何引擎所提供造型功能有限, 因此限制了其造型能力。

在以上系统中, 当一个客户进行模型变换时均采用了某种令牌传递策略, 使其他客户只能观察或查询被置锁的产品模型。选择这种策略主要有两个原因: (1) 在每个操作后实时地更新每个客户端的模型, 在 Internet 环境下受网络带宽限制; (2) 可以避免同时提交的多个特征操作间的潜在冲突。

本文在 CSCW 的架构下^[1], 结合基于细胞元表示的语义特征造型系统^[2], 在哈尔滨理工大学计算机应用技术研究所自行开发的一个基于语义特征的造型系统 HUSTCAID 的开发基础上, 采取了传输造型操作命令的策略, 提出了用操作特征管理器 (operated features manager, OFM) 解决协同设计中的实时数据交换和特征语义一致性问题。

1 基于细胞元表示的语义特征造型系统

基于细胞元表示的语义特征造型系统采用完全语义约束定义特征和构建模型^[3], 能够全面地满足工程设计人员的各种意图, 但是它较之以前的基于参数、基于历程的特征造型要复杂得多。它在底层采用特征依赖图 (FDG)^[2] 保存数据模型, 中层采用细胞元模型作几何模型, 在用户层采用视图模型表示。其体系结构如图 1。

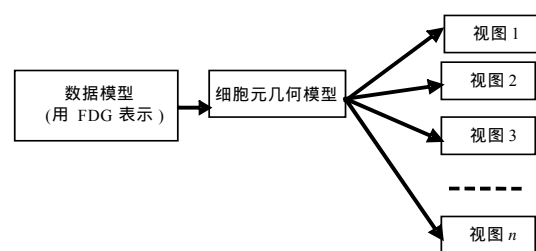


图 1 体系结构

2 传输造型操作层次模型

在现有的 CSCW 系统中, 实现 CAD 系统协同工作的方法一般是视频会议或应用共享, 通过图像共享的方式实现“你见即我见” (WYSIWIS), 这类方法的缺点是协作用户不能直接操作 CAD 模型。在 CAD 模型中包含了大量的高层设计信息, 用户希望能够直接在 CAD 模型上进行实时、在线的协作

基金项目: 国家自然科学基金资助项目 (60173055); 安徽省教育厅自然科学基金资助项目 (2006KJ081B)

作者简介: 王远志 (1977 -), 男, 讲师、硕士, 主研方向: 计算机图形学; 王少威, 讲师、硕士; 孙立铸, 教授、博士生导师

收稿日期: 2006-08-24 **E-mail:** wangyuanzhi1@sohu.com

讨论,并且能够实时地修改模型,在紧密耦合的协同设计中尤其需要这样的功能^[4],协同造型设计的关键问题是如何解决分布式网络环境下协同CAD系统之间的实时数据交换,针对这个关键问题,本文提出了传输造型操作的方法实现协同设计产品模型。

Rafael Bidarra等提出基于细胞元表示的语义特征造型系统基于Web的协同设计^[5],但没有给出具体方案和算法;韩桂鲁等提出泛命令的思想解决网络传输量^[6]问题,但其特征语义一致性不够稳定。

为了保证较低的网络传输负荷和保持模型同步性,在HUSTCAID系统中,本文参照文献^[7]提出的CoDesign体系结构,采用传输造型操作的方法实现协同设计产品模型。即在分布的每台计算机上运行一个HUSTCAID系统的实例进程,每个进程操作并维护一个本地FDG(特征依赖图)。FDG包含了所有的特征实体信息以及它们之间的联系,存储了的复现细胞元几何模型的所需数据,用户可以根据本地FDG得到自己想要的视图,只需实现FDG同步。为此,只要保证某一进程的本地FDG操作能够被传输到其它进程并接收执行,在进程之间通过网络进行高层次通信,即实时传输造型操作,如图2所示,这种通信遵循CORBA规范。它具有以下优点:

(1)网络传输的不是数据量庞大的图像信息,而是具有高层语义的特征造型命令,数据量小、速度快、容易实现实时协同造型。

(2)参加协同设计的人员可以自由地根据FDG得到所需要的视图,灵活性强。

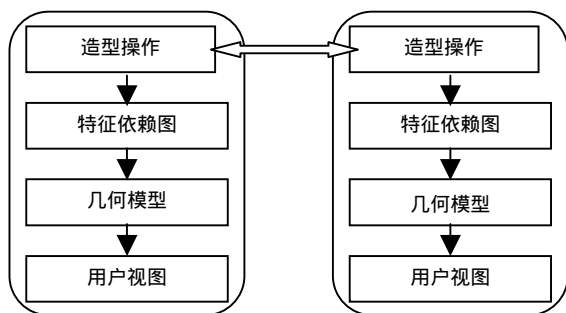


图2 高层通信

3 协同设计实现

在HUSTCAID协同设计中,模型是由多个工程设计人员按照各自的意图设计而成,每个工程设计人员在各自的计算机上运行HUSTCAID系统,各个人员相互协商,完成任务。

3.1 成员

成员表示在小组设计期间一个维护本地FDG的进程。本地FDG包含有各自的特征集合。每一个特征将被分配一个唯一的特征标识符,每一个特征包含若干元素,其中元素(Element)指的是特征的组成部分,如约束、参数、几何元素等。随着操作的不断执行,成员状态在发生改变,假设小组有 n 个成员,本文用向量 $M_k=(m_1, m_2, \dots, m_n)$ 来表示成员 $k(k=1, 2, \dots, n)$ 的状态,其中 $m_i(i=1, 2, \dots, n)$ 是整数,表示已经执行来自成员 i 的操作的个数。

3.2 造型操作

造型操作指成员所执行的动作,是由成员对用户的实际操作进行转化而得到的。在本系统中,造型操作是作为事务来执行的,并利用加锁机制来实现它所维护的特征对象的互斥访问。在FDG里的操作主要有以下几种:特征的插入,修

改和删除,约束的插入,修改和删除,即INSERT、UPDATE和DELETE。每一个操作可以用三元组 $OP<I, M_i, F>$ 表示,其含义是由成员 i 在其状态为 M_i 时所生成的操作 OP , $OP=INSERT|DELETE|UPDATE$,操作的特征是 F ,为方便起见,用 F_{OP} 表示 OP 的操作特征。在这里,不同操作类型的操作特征的含义是不同的,如下:

$$F_I = \{f | f \text{ 是被插入特征所依赖的特征}\}$$

$$E_I = \{e | e \text{ 是 } F_I \text{ 中被依赖的元素}\}$$

$$F_U = \{f | f \text{ 是被修改的特征}\}$$

$$E_U = \{e | e \text{ 是 } F_U \text{ 中被修改的元素}\}$$

$$F_D = \{f | f \text{ 是被删除的特征或 } G(f)\}$$

其中, $G(f)$ 表示那些依赖被删除特征 f 的特征,习惯地将其称为客特征。在FDG里,很容易找到这些操作的对象。

3.3 小组会话

小组会话指的是有限成员在有限时间对FDG进行有限操作的一段过程,即设计小组实现协同设计的一段过程。小组会话开始时,各成员处FDG状态一样, M_1, M_2, \dots, M_n 都为零向量;有效时, $M_1 = M_2 = \dots = M_n$ 。

4 特征语义一致性的维护

特征语义一致性是指在进行小组会话后所有成员所维护的模型状态是一样的,其不一致性是由于进程接收的操作命令执行不统一而引起的,因此其关键就在于如何控制操作的执行。对此,史美林等^[7]根据操作优先关系和重叠关系进行相应的操作,但没有解决插入与插入、插入与修改以及插入与删除的冲突;一些用令牌来控制操作权的方法,严格地限制了操作的执行顺序,使操作串行执行,速度很慢。在HUSTCAID系统中,模型是由FDG决定的,保证FDG的一致即可。下面结合HUSTCAID系统,分析其特征语义不一致的原因,根据操作相容性,用操作特征管理器来控制操作的执行。

4.1 操作的相容性

操作的相容性指的是两个或两个以上的操作无论以何种顺序执行都能保证FDG的结果唯一性,而相容性有利于FDG的特征语义一致性。下面针对3种操作:INSERT, UPDATE, DELETE,分析它们之间的相容性。

(1)INSERT和INSERT(分别用 I_1 和 I_2 表示)

当 $F_{I_1} \cap F_{I_2} = \emptyset$ 时, I_1 和 I_2 两个操作互不影响,互相独立,二者的执行顺序不影响最终的模型结果,此时它们相容。

当 $F_{I_1} \cap F_{I_2} \neq \emptyset$ 时,还不能立即肯定 I_1 和 I_2 是否相容,本文需要进一步判断 $E_{I_1} \cap E_{I_2}$ 是否为空,即 I_1 所插入的特征和 I_2 所插入的特征是否依赖同一特征实体同一参考元素上。如果 $E_{I_1} \cap E_{I_2} = \emptyset$ 时,由于二者的操作是没有关联的,因此相容;如果 $E_{I_1} \cap E_{I_2} \neq \emptyset$,由于稍后操作可能被先前操作阻碍,因此 I_1 和 I_2 的冲突的发生率比较高,为了保证模型的安全性,设定它们不相容。

(2)INSERT和UPDATE(分别用 I 和 U 表示)

这2个操作的相容性与上面类似,当 $F_I \cap F_U \neq \emptyset$ 时,二者相容;当 $F_I \cap F_U = \emptyset$ 时,若 $E_I \cap E_U = \emptyset$,二者相容,若 $E_I \cap E_U \neq \emptyset$,二者不相容。

(3)INSERT和DELETE(分别用 I 和 D 表示)

当 $F_I \cap F_D = \emptyset$ 时,由于二者互不影响,因此相容;当 $F_I \cap F_D \neq \emptyset$,若 I 执行在前, D 执行在后,将会导致插入的部分特征集 $F_I \cap F_D$ 会马上被删掉, I 执行的功能无效,若 D 执行在先, I 后执行,则 I 所插入的特征没有所依赖的主特征, I 无法进行,

认为当 $F_i \cap F_D \neq \emptyset$ ，这两个操作不相容。

(4)UPDATE 和 DELETE(分别用 U 和 D 表示)

当 $F_U \cap F_D \neq \emptyset$ 时，由于二者互不影响，因此相容；当 $F_U \cap F_D = \emptyset$ ，在执行过程当中，采用合适的策略，可以使它们相容。若 U 执行在 D 之前， F_U 先根据 U 操作修改， F_D 再被删除，结果是 F_U-F_D 维持为被 U 修改的状态， F_D 全被删掉；若 D 在 U 之前执行，先把 F_D 删掉，然后只对 F_U-F_D 部分特征根据 U 修改，其结果显然和前一种情况一样、和预期的一致。这两种操作在任何时候都相容。

(5)UPDATE和UPDATE(分别用 U_1 和 U_2 表示)

当 $F_{U1} \cap F_{U2} \neq \emptyset$ 时，由于各自修改不同特征，因此它们的执行顺序不会影响最终的结果，是相容的。

当 $F_{U1} \cap F_{U2} = \emptyset$ 时，也需进一步检查，才能确定它们的相容性。若 $E_{U1} \cap E_{U2} = \emptyset$ ，即虽然它们修改的是同一特征，但不是同一个元素，两个操作仍然相容；若 $E_{U1} \cap E_{U2} \neq \emptyset$ ，即修改的是同一元素，很明显不同的顺序会导致不同的结果，此时二者不相容。

(6)DELETE和DELETE(分别用 D_1 和 D_2 表示)

当 $F_{D1} \cap F_{D2} = \emptyset$ 时，显然可以相容；当 $F_{D1} \cap F_{D2} \neq \emptyset$ ，若 D_1 执行在前，先删除 F_{D1} ，再删除 $F_{D2}-F_{D1}$ ，最终结果是删除了 F_{D1} F_{D2} ，若 D_2 执行在前，结果一样。对于两个删除操作，任何时候它们都相容。

4.2 操作特征管理器控制模型

操作特征管理器(OFM)运行在集中服务器上，负责统一管理操作特征，上面记录了当前可执行操作的操作特征 F_{OP} 。在小组会话的过程当中，各成员可以自由地向OFM提出操作请求，成员的操作都需OFM根据上述操作相容规则来检查批准，凡是被批准的操作都是相容的，这些操作将被各个成员执行；凡是可能发生冲突的操作都被拒绝，直到OFM中的相关操作执行完毕，才被OFM接收处理。OFM批准的操作所形成的操作集合可以在任何成员处以任何顺序执行，且能保证模型结果的一致性，并且不同的操作在各个成员处完全可以并行执行，从而在整个协同设计过程中能缩短操作流的执行周期，会话效率上升，速度提高。为了能确认某一操作是否被所有成员执行，在

OFM 中用向量 V_{OP} 来标志 OP 的执行情况， $V_{OP}=(v_1, v_2, \dots, v_n)$ ，每位初始化为 0，表示未被任何成员执行，当成员 $i(i=1, 2, \dots, n)$ 执行了操作后，就让 $v_i = 1$ 。当 V_{OP} 的每一分量都为 1 时，表示该操作成功地被所有成员执行。一旦该操作被执行后，在OFM中 F_{OP} 就被释放，从而可以让其它不相容的操作被OFM批准并得以执行。在小组会话期间，都是围绕OFM进行的，假设小组会话没有成员中断，会话完毕后， $M_1 = M_2 = \dots = M_n$ ，各成员FDG的特征语义一致性将得到有效维护(图 3)。

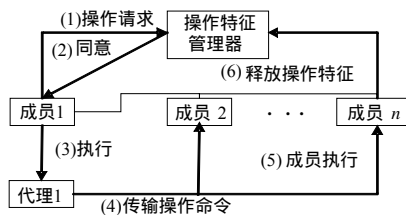


图 3 OFM 模型

5 实例分析

下面以成员 1 的一个修改操作为例分析造型操作的执行过程，可参见图 3。

(1)成员 1 向OFM提出UPDATE操作请求，把所要修改的特征 F_U 提交给OFM。

(2)OFM根据操作的相容性规则，把 F_U 和OFM中的其它操作特征集进行集合运算，检查该操作和其他操作的相容性；如果相容，则把 F_U 添加到OFM，并发送消息同意成员 1 执行 UPDATE 操作并转至(3)，否则拒绝UPDATE操作并结束成员 1 的操作请求。

(3)成员 1 执行修改操作并确认。

(4)代理接收来自成员 1 的确认消息后，把成员 1 的操作命令通过网络传输给其他成员。

(5)其它成员接收命令后，立即执行；每个成员执行完 UPDATE 操作命令后，把各自状态向量的分量 m_i 加 1，并通知OFM。

(6)OFM接收到来自成员 i 的通知后， V_U 的分量 $v_i = 1$ 。当 V_U 的每个分量都为 1 时，表示该操作成功地执行，OFM 释放 F_U 。

经过以上 6 个步骤，成员 1 的 UPDATE 操作就被成功地执行。

6 结论

用 OFM 控制成员之间的操作，使得操作流的执行不再发生冲突，从而解决了协同设计中的特征语义一致性问题。本方法无需复杂的算法，在 Windows2000 平台上和网络环境下，用各种程序语言教能开发，在 HUSTCAID 系统中的用 VC++ 得到了初步实现和应用，没有冲突现象发生，具有可行性。但是，这种方法也有些局限性，需要进一步研究解决：

(1)要求 2 个 CAD 系统的造型方法一致，只有造型方法一致时，才能实现高层特征操作数据的交换。

(2)当模型的特征间关系耦合程度非常低时，OFM 中的相容操作将较少，效果就非常有限。

参考文献

- 1 Meilin S. CSCW: Computer Supported Cooperative Work[J]. China Journal of Communications, 1995, 16(1): 55-61.
- 2 孙东光, 李隆春, 郭连水, 等. CAD 系统语义特征模型的研究与实现[J]. 计算机集成制造系统——CIMS, 2002, 8(3): 193-196.
- 3 许宇荣, 孙守迁, 潘云鹤. 基于约束的智能协同设计系统模型的研究[J]. 中国机械工程, 2002, 13(4): 333-336.
- 4 刘锦兴, 秦叶, 李荣彬, 等. 基于信息网络的异地协同设计与制造系统研究[J]. 中国机械工程, 1999, 10(8): 882-884.
- 5 Bidarra R, Bronsvort W F. Collaborative Modeling with Features[C]//Proc. of the 21st Computers and Information in Engineering Conference. 2001: 535-545.
- 6 韩桂鲁, 陈立平, 范朋旺. 网络协同三维零件设计技术及系统研究[J]. 计算机辅助设计与图形学学报, 2002, 14(6): 574-577.
- 7 Shi Meilin, Yang Guangxin. CoDesign: Prototype of a Cooperative Design Support System[J]. Journal of Tsinghua University, 1998, 38(1): 30-35.