

基于特征优先关系的冲突消解方法^{*}

丁 博, 孙立镔

(哈尔滨理工大学 计算机科学与技术学院, 哈尔滨 150080)

摘 要: 提出了一种基于特征优先关系的冲突消解方法, 该方法通过制定设计人员的静态优先级和特征操作的动态优先级, 可以有效地预防、检测和消解冲突, 维护了协同模型的一致性。同时, 还采用了负载平衡技术, 保证了协同建模的效率。

关键词: 同步协同设计; 冲突消解; 特征优先关系; 负载平衡技术

中图分类号: TP391 **文献标志码:** A **文章编号:** 1001-3695(2009)01-0366-03

Conflict resolve method based on feature precedence relation

DING Bo, SUN Li-juan

(College of Computer Science & Technology, Harbin University of Science & Technology, Harbin 150080, China)

Abstract: This paper proposed a conflict resolve method based on feature precedence relation, which could prevent, detect and resolve conflict effectively, maintain consistency of the models by the way of making static priority of designers and dynamic priority of feature operation. Besides, this paper used load balancing technology to ensure the efficiency of collaborative design.

Key words: collaborative design; conflict resolve; feature precedence relation; load balancing technology

CSCW^[1]的基本含义是在计算机技术支持的环境中, 一个群体协作地完成一项共同的任务。由于 CSCW 是一种基于网络的分布式系统, 在协同设计过程中就必然存在并发冲突^[2], 冲突破坏了模型数据的完整性和一致性, 使系统无法正常运行。由于 CAD 模型具有复杂度和耦合度高等特点, 给冲突控制带来了难度。冲突控制^[3]的目的是对并发操作进行有序控制, 正确协调系统中出现的并发操作。

本文针对本实验室实现的混合式同步协同平台提出了一种基于特征优先关系的冲突消解方法。该方法进一步强调管理在协同建模时的重要性, 在协同设计开始前, 先将所要完成的任务划分成若干个子任务, 确定每个设计人员应完成的子任务, 根据每个设计人员担任的任务不同为其分配不同的角色; 并通过角色机制为其分配相应的静态优先级, 角色机制是并发控制的基础。此混合式系统支持设计人员传输特征、零件或装配体, 并采用特征依赖图^[4]记录模型构造的全过程, 通过细胞元^[5]语义特征造型技术^[6]中特征间的依赖关系确定了特征动态优先关系, 使并发冲突得到有效的控制。

1 相关知识介绍

1.1 并发冲突的种类

1) 因果冲突 是指系统在没有执行因操作前, 先被要求执行果操作。此冲突有可能是由网络延时造成的。

2) 意图冲突 是指当操作在一个站点生成后被传送给其他站点进行同步执行, 但在这个操作到达某个站点执行之前这个站点的对象模型已经被本地操作或其他远程并发操作修改, 因此当这个操作执行时它所基于的模型上下文环境已经发生改变。这样操作执行的结果可能与操作在本地站点的执行结果大相径庭, 从而不满足设计者的原有意图。

3) 分歧冲突 是指不同设计者对模型的同一特征的同一属性共同关注而产生的。这些操作无论谁先执行谁后执行对象模型都能且仅能体现其中一个操作的意图, 其他操作的执行结果无法表现出来。

1.2 并发控制方法

1) 并发冲突防止 是通过某种规则对用户操作进行一定限制, 使他们不会发生并发冲突。

2) 并发冲突检测与解决 是通过一定的操作冲突判别规则对系统中的并发操作进行冲突判断, 然后再依据一定规则来对发生冲突的操作进行协调处理。

本文提出的基于特征优先关系的冲突消解方法, 充分分析了冲突产生的原因, 结合了以上两种基本并发控制方法, 可以有效地预防、检测和消解冲突。

2 系统框架

本实验室研发的混合式平台由一个主站点和若干从站点组成。主站点不从事任何构建模型的工作, 它只用来接收和发送从站点的建模指令, 验证建模指令的合法性, 并对各建模指令进行合理的调度, 维护整个建模过程; 从站点的任务是构建模型。主站点由四个模块组成, 即消息控制模块、权限模块、调度模块和负载平衡模块, 如图 1 所示。

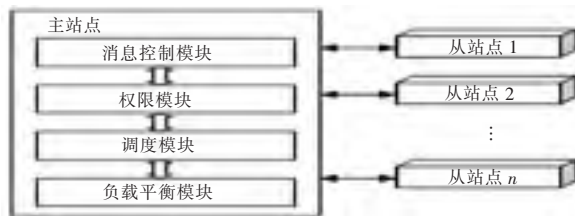


图 1 混合式平台体系结构

收稿日期: 2008-03-31; 修回日期: 2008-05-29 基金项目: 国家自然科学基金资助项目(60173055)

作者简介: 丁博(1983-), 女, 博士研究生, 主要研究方向为计算机图形学与 CAD(dingbo@hrbust.edu.cn); 孙立镔(1944-), 男, 教授, 博导, 主要研究方向为计算机图形学与 CAD、主持完成三项国家“九五”重点科技攻关项目。

1) 消息控制模块 用来验证并接收从站点传送来的建模数据。验证一个操作是否可执行,通过检查操作对象特征的直接依赖特征在站点中是否均已存在来判断。若验证成功,则说明此操作对象特征的直接依赖特征已存在,主站点便发送验证成功消息给此从站点;若失败,主站点则发送验证失败的消息给此从站点,并说明验证失败的原因。消息控制模块监控任意操作执行情况的全过程,当某一从站点传送的建模数据经验证成功并被其他各从站点接收时,消息控制模块将发送传送成功的消息给此从站点。若发生传送超时等异常状况时,消息控制模块将发送消息给此从站点,让其重新传送相关建模数据。

2) 权限模块 采用角色机制,其作用是获取用户的静态优先级数。系统根据设计人员所担任的不同角色事先划定相应的静态优先级。角色作为中间桥梁将设计人员与权限联系起来,一个角色与权限关联可以看做是该角色拥有的一组权限的集合,与设计人员关联又可以看做是若干具有相同身份的设计人员的集合。一个登录于某一系统的设计人员,可以通过他所具有角色的权限来确定其可访问的系统资源和对系统资源可以进行的操作。角色机制使若干设计人员可以共同承担一个角色,当其中一名设计人员因某种原因不能继续工作时,也不会影响工作进程。由于本文已事先确定不同角色的设计人员应完成的子任务,要求设计人员不能越权对模型进行操作,在修改某一特征时,只有该特征的创建者才能对其进行修改操作,拒绝其他设计人员对其的修改请求,有效避免了分歧冲突的产生。

3) 调度模块 为每一个子站点创建一个子中性建模队列,每个子站点的建模操作按时间顺序依次放入到各自的子中性建模队列中;然后根据特征优先关系将待执行的操作放入中性建模队列中。

4) 负载平衡模块 采用负载平衡技术对调度模块中的各子中性建模队列进行实时监控,根据各子中性建模队列中的待处理操作数量的多少进行任务调度。当设计人员间不存在并发操作时,便通过负载平衡技术先执行待处理操作数量多的子中性建模队列中的操作,此技术使每个设计人员的建模操作能够尽量快地被执行,不会因长期等待而影响他们的设计积极性。由于本混合式系统支持设计人员传输特征、零件和装配体,就可能出现设计人员在某一时间段内,只是设计模型,而没有传输设计完成的模型,使主站点处于空闲状态。这时,通过负载平衡技术,系统将从静态优先级最高的站点开始,向各站点发送消息,让它们传输建模数据。

3 特征间的优先关系

3.1 特征依赖集

特征依赖图是有向无环图,图中的一个节点对应模型中的一个特征,存在有向边连接的两个节点表示节点对应的两个特征存在直接特征依赖关系,有向边的始点是父特征,终点(即箭头所指方向)是子特征,子特征因父特征的存在而存在。若两个特征间不存在依赖关系,则称这两个特征互为独立特征。图 2 为一特征依赖图实例。

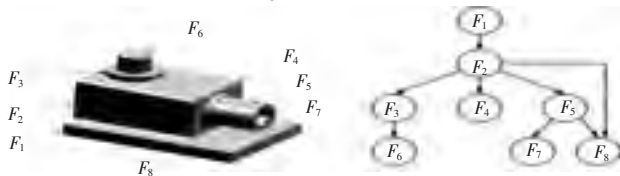


图 2 特征依赖图

对于模型中的所有特征构成的集合 S ,假设特征集 S 中有

任意三个特征 F_1, F_2, F_3 ,则可知它们满足下面的三个性质:

- a) 传递性,如果 F_1 优先 F_2, F_2 优先 F_3 ,则 F_1 优先 F_3 。
- b) 单向性,如果 F_1 优先 F_2 ,则 F_2 不优先 F_1 。
- c) 自反性, F_1 不优先于自己。

如果特征 F_1 的构造直接引用了特征 F_2 ,或与特征 F_1 有直接的约束关系,则称 F_2 是 F_1 的直接依赖特征。同样,如果特征 F_1 是通过特征依赖的传递性而对特征 F_2 产生的依赖关系,则称 F_2 是 F_1 的间接依赖特征。

特征依赖关系描述了特征间的工程语义关系,体现了一种特征间的相互影响特性,即特征的变动只影响依赖于它的特征,而与其互为独立特征的特征是不受直接影响的。当在原有模型中添加、修改、删除特征时只会影响到它所依赖的父特征的拓扑结构,从而可能影响到依赖于该父特征的所有特征。因此,对于任意特征 F_1 的依赖特征可能有多个,包括直接依赖特征和间接依赖特征,这些特征的集合构成了特征依赖集 $dependSet(F_1)$ 。

在进行同步协同建模时,如果要对特征 F_1 进行修改,可以通过遍历特征依赖图,确定特征操作所影响的特征依赖集 $dependSet(F_1)$ 。由于仅需对 $dependSet(F_1)$ 中的特征进行相应的修改,而不用重构整个原有模型,极大提高了建模效率。

3.2 动态优先级的确定

基于细胞元表示的语义特征模型构建造型时,其特征间的优先次序是动态改变的,可以根据特定阶段特征之间的依赖关系来确定优先次序,因此,每次特征操作的优先次序也应随着特征间的优先次序的改变而动态改变,每次进行特征操作时均要对特征间的优先性进行分析。本文就依据语义特征模型的这一特性来定义协同建模过程中特征操作的动态优先级。

根据特征依赖图及特征间的优先关系,特征操作时的动态优先级应遵守以下三条原则:

- a) 特征依赖图中的每一条边表明了模型中两个特征之间的优先关系:若 F_2 依赖于 F_1 ,则 F_1 的动态优先性高于 F_2 。
- b) 属性不同的两个特征(一个为正特征,一个为负特征),原先互不相交(互为独立特征)。在经过修改后,两特征相交,若重叠是 F_2 造成的,那么 F_1 动态优先级高于 F_2 。
- c) 属性相同的两个特征 F_1 和 F_2 (同为正特征或同为负特征)。经过修改特征 F_1 后,两特征相交,生成一个新的细胞元。则新生成的细胞元在特征模型中的动态优先级调整为未经修改时的特征 F_2 的优先级。

4 冲突消解方法

定义 1 操作 M 。 $M = (M_{id}, F_{id}, U_{id}, sta_prio, FS, Type, dyn_prio, dependSet(F_{id}), Attribute, T_s, T_e)$ 。其中: M_{id} 是设计人员 U_{id} 在系统中作用于特征 F_{id} 之上任一操作的惟一标志符; sta_prio 是设计人员 U_{id} 的静态优先级; FS 是该操作的类型; $Type$ 是该操作的子类型; dyn_prio 是该操作的动态优先级; $dependSet(F_{id})$ 是该操作对象特征的特征依赖集; $Attribute$ 是该操作涉及的所有属性列表; T_s 是操作开始时刻; T_e 是操作结束时刻。

定义 2 并发冲突。对于协同设计中的任意两个操作 M_i 和 M_j ,它们的操作对象特征分别是 F_i 和 F_j ,如果这两个操作同时满足如下条件:a) $M_i \times U_{id} \neq M_j \times U_{id}$; b) $[M_i \times T_s, M_i \times T_e] \cap [M_j \times T_s, M_j \times T_e] \neq \emptyset$; c) $dependSet(F_i) \cap dependSet(F_j) \neq \emptyset$ 。这两个操作存在并发冲突。

本文通过自定义的静态优先级和动态优先级,提出了基于

特征动态优先关系的冲突消解方法。其步骤如下:

a) 对于任意两个并发操作 M_i 和 M_j , 首先比较这两个并发操作的操作对象是否为同一特征, 若为同一特征, 则必有一个操作是越权操作, 应放弃该操作。其次比较两者的特征依赖集, 如果 $\text{dependSet}(F_i) \cap \text{dependSet}(F_j) \neq \emptyset$, 则存在冲突, 转 c); 否则转 b)。

b) 比较并发操作 M_i 和 M_j 的静态优先级, 若两者的静态优先级不同, 且 $M_i \times \text{sta_prio} > M_j \times \text{sta_prio}$, 则按静态优先级将操作 M_i 和 M_j 依次放到中性建模队列中并等待执行; 若两者的静态优先级相同, 转 d)。

c) 比较并发操作 M_i 和 M_j 的静态优先级, 若 $M_i \times \text{sta_prio} > M_j \times \text{sta_prio}$, 则执行操作 M_i , 放弃操作 M_j ; 若两者的静态优先级相同, 转 e)。

d) 比较两者的动态优先级, 如果 $M_i \times \text{dyn_prio} > M_j \times \text{dyn_prio}$, 将按动态优先级将 M_i 和 M_j 操作依次放到中性建模队列中并执行。

e) 比较两者的动态优先级, 如果 $M_i \times \text{dyn_prio} > M_j \times \text{dyn_prio}$, 则将动态优先级高的操作 M_i 放到中性建模队列中并等待执行, 放弃操作 M_j 。

由于本系统支持设计人员一次性传输特征、零件或装配体, 对零件和装配体只需提取其父特征, 按以上步骤执行便可。

5 系统实现

本文以一台主服务器和三个协同建模站点为例, 验证了基于特征优先关系冲突消解方法的有效性(图 3)。其中站点 1 与 2 的静态优先级相同, 站点 1 大于 3 的静态优先级。

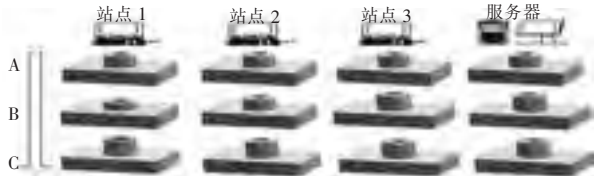


图 3 冲突消解方法有效性的验证实例

(上接第 365 页)基本的轮廓波变换^[5]。图 4(a)为某海岛图像的一角;(b)是分解层数为[0,3,3,4,4]的轮廓波变换的逼近效果;(c)为小波变换的逼近效果;(d)为本文提出的混合波变换的逼近效果。从图中可以看出,本文提出的混合变换逼近的视觉效果远远优于小波变换,略逊于轮廓波变换(轮廓波变换具有更多的“人为线段”),但是由于其计算时间比轮廓波变换短约 40% 并具有较小的冗余度,使得该变换更容易满足实时处理和图像压缩的要求。

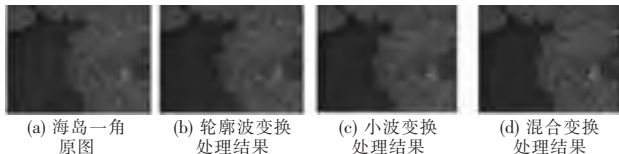


图 4 海岛一角逼近效果图(100:1)

4 结束语

本文提出了一种基于轮廓波变换与小波变换的混合变换。该变换在较粗糙的尺度上使用小波变换,在更细节的尺度上使用轮廓波变换,是一种固定基描述方法。实验结果表明,在大约 100:1 的压缩率下,使用该变换可以有效地逼近原始海岛图像。其逼近效果优于小波变换,其冗余度略低于轮廓波变

时间流水线 A 时刻是初始模型,各站点的模型均是一致的。B 时刻站点 1 将圆柱体高度降低,站点 3 将圆柱体高度增加,虽然站点 1 高于站点 3 的静态优先级,但该圆柱体是由站点 3 创建的,系统判定站点 1 发生越权操作,放弃执行,只执行站点 3 的操作。C 时刻站点 1 将盲孔的半径增大,而站点 2 将模型的一条棱边倒圆,经验证这两个操作的特征依赖集并不相交,即不存在并发冲突。由于两者的静态优先级相同,比较其动态优先级,站点 2 大于站点 1 动态优先级,将这两个操作依次放入中性建模队列中,服务器先执行站点 2 的操作。

6 结束语

并发冲突的预防、检测和消解在同步协同设计过程中是至关重要的。本文采用基于特征优先关系的冲突消解方法,通过特征依赖图记录模型构造的全过程,制定设计人员的静态优先级和特征操作的动态优先级,有效维护了协同模型的视图一致性,并以高层语义造型命令为传输载体,通过负载均衡技术,充分利用了网络资源,减少了网络的负荷,支持了协同设计的有效进行。

参考文献:

- [1] MORTEN O F, KRISTINA L, MARTIN B, *et al*. Physical and virtual tools: activity theory applied to the design of groupware [J]. *Computer Supported Cooperative Work*, 2002, 11(1-2): 153-180.
- [2] IONESCU M, MARSIC I. Tree-based concurrency control in distributed groupware [C] // *Proc of ACM Conference on Computer Supported Cooperative Work*. Norwell, MA: Kluwer Academic Publishers, 2003: 329-350.
- [3] 余春燕, 侯宏仑, 吴明晖. 协同设计中以实体为中心的并发操作控制机制 [J]. *计算机辅助设计与图形学学报*, 2004, 16(9): 1289-1295.
- [4] BIDARRA R, BRONSVOORT W F. Semantic feature modeling [J]. *Computer Aided Design*, 2000, 32(3): 201-225.
- [5] BIDARRA R, NEELS W J, BRONSVOORT W F. Boundary evaluation for a cellular model [C] // *Proc of Computers and Information in Engineering Conference*. Chicago: [s. n.], 2003: 255-265.
- [6] LIA W D, ONG S K, JERRY Y H, *et al*. Feature-based design in a distributed and collaborative environment [J]. *Computer Aided Design*, 2004, 36(9): 775-793.

换, 计算时间和硬件复杂程度低于基本轮廓波变换, 比轮廓波变换更适合于数字图像实时处理的场合以及要求较高压缩比的情况。

参考文献:

- [1] MALLAT S. A wavelet tour of signal processing [M]. Orlando: Academic Press, 1999.
- [2] CANDES E J, DONOHO D L. Ridgelets: a key to higher-dimensional intermittency [J]. *Philosophical Transactions of the Royal Society of London Series A: Mathematical and Physical Sciences*, 1999, 357(1760): 2495-2509.
- [3] PENNEC E L, MALLAT S. Sparse geometric image representation with bandelets [J]. *IEEE Transactions on Image Processing*, 2005, 14(4): 423-438.
- [4] COHEN A, MATEI B. Compact representation of images by edge adapted multiscale transforms [C] // *Proc of IEEE Int Conf on Image, Special Session on Image Processing and Non-linear Approximation*. Piscataway, NJ: IEEE Press, 2001: 8-11.
- [5] DO M N, VETTERLI M. The contourlet transform: an efficient directional multiresolution image representation [J]. *IEEE Transactions on Image Processing*, 2005, 14(12): 2091-2106.
- [6] PHOONG S M, KIM C W, VAIDYANATHAN P P, *et al*. A new class of two-channel biorthogonal filter banks and wavelet bases [J]. *IEEE Transactions on Signal Processing*, 1995, 43(3): 649-665.
- [7] 陈新武, 刘金根. 基于 IEEE Std 1149.1—2001 标准的 TAP 控制器设计 [J]. *信阳师范学院学报: 自然科学版*, 2006, 19(4): 452-454.