

# 分布式参数化路径计划技术初探

屠立忠<sup>1</sup>, 季平<sup>2</sup>, 左敦稳<sup>1</sup>, 王珉<sup>1</sup>

(1. 南京航空航天大学机电工程学院, 南京 210016; 2. 香港理工大学制造工程系, 香港九龙红磡)

**摘要:** 在讨论了现有分布式虚拟现实系统的缺点的基础上, 提出了一种以参数路径计划技术为核心的虚拟现实系统。在这种虚拟现实系统中, 共享及管理的对象路径是以时间为参数的线性或高阶函数, 各个远程分布系统用标准时钟服务器来同步。该方法可以解决广域网络虚拟现实系统的长时间延迟问题, 预先进行干涉检查操作并大幅度减少此时网络上的信息量, 减少甚至消除显示移动滞后, 另外还具有增加捕捉用户动作的平滑度的能力。

**关键词:** 虚拟现实; 参数化路径计划; 同步时钟; 网络延迟; 干涉检查

**中图分类号:** TP391.9 **文献标识码:** A

## An Approach to the Distributed Parametric Path Planning

TU Li-zhong<sup>1</sup>, JI Ping<sup>2</sup>, ZUO Dun-wen<sup>1</sup>, WANG Min<sup>1</sup>

(1. School of Electromechanics Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, 210016;

2. Department of Manufacturing Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong)

**Abstract:** Based on current distributed virtual reality systems, a new approach to the distributed parametric path planning is put forward in this paper. In this virtual environment, paths of objects are linear or high-order functions of time, and all remote workstations are synchronized by standard clock server. The problem of long communications over wide area network can be resolved. In this system, information traffic can be cut down with cache ahead collision detection, and temporal display shifts can be reduced and even removed. This system also has the capability of increasing the level of smoothness when capturing user movement.

**Keywords:** virtual reality; parametric path planning; synchronized clock; network delay; collision detection

## 前言

目前, 虚拟现实技术研究工作重点放在硬件技术、计算平台、产品的模型管理和仿真技术的开发上。最近, 也出现了在网上进行分布处理的新思路, 它们采用面向对象的模型管理方法实现了对象模型信息的传递, 这种思路都是通过消息广播的方式将系统中的更改信息通知所有的任务工作站系统。这种方法是以为网络足够的频带宽度和系统快速响应为基础的, 在目前情况下, 远距离操作很难实现。而使用参数化路径设计方法, 数据同步和时间延迟的问题就能迎刃而解。本文将在讨论现有分布式虚拟现实系统的缺点的基础上, 提出以参数化路径计划技术为核心的虚拟现实系统。

## 1 常用的虚拟现实对象模型管理方法

为了便于理解虚拟现实系统的参数化路径设计方法, 我们先来讨论一下常用的虚拟现实模型管理方法。

### 1.1 共享模型方法

通常系统网络中每一台参与仿真计算的计算机都有一套完整的虚拟现实模型, 这是最大限度降低网络负担的最佳方案, 因为模型的读取是最频繁的操作, 这种操作可以在本地实施而不必依赖网络, 因而对网络的频带要求减低。

数据库的更改只是偶而为之的操作。通常在两种情况下才会出现: 一是在网络上运行应用软件包, 一是用户交互操作模型。当一个系统对模型作出了修改时, 为了保持每个系统中的虚拟模型的一致性, 包括远程系统在内的其他系统都会得到维护模型的通知。这种更改请求通常是以在网络上广播的方式发出, 其他系统从网上拾取更改信息, 对它们本机的虚拟模型的备份作出更改。

这种方法对于系统采用宽频网络而且网络延迟时间远小于仿真循环周期时, 系统可以正常工作。但是在一些同步系统中, 在下一仿真实施前机器必须物理停机, 直到从其他机器来所有的更改都完成为止。言下之意, 长延迟网络的系统, 循环周期将被网络延迟时间所控制。而在异步系统中, 长延迟网络可能引起远距离机器模型更改的不一致或不完整, 因此对于两个试图远距离合作的人来讲确实存在着问题。

## 1.2 时间离散方法

很多系统是基于逐帧方式实现速度和状态的变化。例如,质子的运动轨迹逐帧地描述成一个具有相对速度的正矢量。换句话说,在模型更新过程中,这个正矢量是通过增加速度分量的方式更新的。通过这个简单的方法实现在虚拟世界中物体的运动,这种修正的迭代技术常常被很多虚拟现实的仿真系统所采用。

不过,在运行仿真程序时,这种简单方法完全依赖于模型的更新时间。当系统要求处理模型的变化时,物体的运动路径计算依赖于系统的加载速度。由于模型更新任务在网上趋向于以不同的速度运行,为了保证模型的一致性,系统必须负责更新物体的位置并且在网上广播这个物体当前的位置。

对于一个匀速运动的物体,网络上必须一直不断地发出广播信息更新它的位置。除此之外,系统中也将有大量的逐帧信息送上网络。最后,由于系统长时间的通讯延迟,物体位置信息的接收远远落后于模型的当前位置。除非在网络延迟是无关紧要的情况下,帧速度是恒定而且同步的,否则这些问题将严重妨碍运动路径计划的完成。而在广域网上,网络延迟是不可避免的。

事实上,假设渲染速度恒定不变,模型位置计算时间和显示该模型时间之间便会出现偏差。在很多情况下,这并不是大问题,但是对仿真来说,有了延迟便不是实时仿真了。

## 1.3 干涉检查

因为大多数系统中采用离散方法,干涉检查必须基于逐帧实施,这需要很大的计算量。通过指定允许干涉物体的边界盒方法及设定层次限制计算量的方法可以减少一部分计算量,但是最基本的检查在每一帧都得进行。检查的复杂程度跟物体和物体相交的层次有着密切的关系,这就是改变系统负载甚至整个系统周期的潜在原因。

## 1.4 用户与物体的交互

用户和虚拟世界中的模型交互活动,是通过身体跟踪系统来操作虚拟世界中的人这种“物体”的。这些“物体”在虚拟世界中代表人的身体的一部分。“物体”的位置是以循环方式读取用户身体的状态后更新的。在对象运动仿真中,当用户的状态变化时,为了保持对象模型的一致性,这些变化也必须在网上广播,而且每一个在网上的用户都必须这样做。如果系统运行的刷新速率不同时,必须使用前述的离散时间方法而潜在的模型不一致问题将不可避免。

# 2 参数化路径计划方法

使用参数化定义路径(简称参数化方法)加上同步时钟系统便可使上述问题迎刃而解。下面详细讨论这种方法。

## 2.1 同步时钟的概念

参数化方法在很大程度上依赖于标准时钟的存在,所有

分布处理平台的实时时钟都与标准时钟同步,为了合理有效,同步间隔时间必须小于网上所有计算机系统的最短刷新时间,最理想的间隔是50Hz(五十分之一秒),因为这一点在实用中很容易达到。

通常采用时钟服务器完成同步过程,当计算机出现时钟滞后时,时钟服务器便发送同步请求。同步请求一般以重复方式发送,当然为了校正近程甚至远程计算机的时钟,请求信号在网上延迟等因素都必须考虑在内以提高准确度。

所有有关对象模型交互操作都必须带有标准时钟标记。

## 2.2 对象的参数化轨迹

有了同步时钟的配合,虚拟世界中的所有运动都编码成参数化形式,参数为时间。对于简单速率情况,造型系统中对象被定义成有一个起始点,起始点方向速度矢量的时间函数和一个终止条件。简单的对象运动可以用线性方程表示,而对象的当前位置可由当前系统的时钟和方向速度矢量计算得到。尽管这种方法要比简单迭代法复杂得多,但是这样可以精确地将时钟校正到当前标准时间。而对于复杂轨迹的对象,可以使用高阶方程或者带有适当时间参数的线性方程表示。

## 2.3 对象路径队列

如果可能的话,可以预先定义好对象的许多路径段元素。在虚拟世界中,预先定义好整个场景,场景中包含详细的设计路径。对于复杂的路径,可以转换成简单路径的队列的方法。通过定义初始条件、路径方程和包含时间参数的端点条件,队列元素可以变成在最低队列元素为基础的超越终点条件的实时时钟函数。

新的路径更新以下面的方式进入队列:首先接收新路径元素,如果新元素的开始时间代码大于后面元素的开始时间代码,那么新的路径元素即被加入队列,而前一个元素的终止时间代码设置成新元素的开始时间代码。使用这种方法可以保存或覆盖保存将来的路径计划并且保持对用户的不可见性。

## 2.4 实现参数化共享模型

在网络上实现这种参数化方法采取的是同早期系统类似的方案。模型还是由带有系统间传递的信息对象,信息传递方式还是和其他系统一样。只不过现在所有的对象增加了位置及其他属性的参数路径堆栈,而每个模型的旋转元素将由放在这些堆栈中的指令更改。同以前的系统一样,每一个计算平台的模型共享环境中都有一份虚拟现实数据库的备份,一个系统对数据库作出的修改将会广播到网上同组的其他成员。

要重点注意的是,在参数化系统中,数据库的任何变化都和时间代码有关,并且都是按压栈的顺序变化,对象位置和方位的变化都只影响路径队列。一个对象的路径可以设计成许多参数化路径段,时间参数是以标准时间为参考,在

网络上广播的新路径元素被所有参与的系统接受并且压进堆栈。通过这种方法可以捕捉到对象与对象、人与对象的交互过程,并显示对象交互过程。

同离散时间方法相比,首先参数化系统一定程度上减少了网络的拥挤,由于系统中有对象路径的完整描述,有现在的状态,也有将来的状态,因而不需要不断地向网络广播。第二,参数化系统中对象的运动是连续的、平滑的,与帧刷新率无关。虽然每个系统独立运行,但是由于它们使用了标准时间参考的参数化模型,每个用户可以用独立的渲染时间准确计算出对象的位置。甚至系统加载过程改变以及帧刷新率降低时,对象的实时位置和运动状态还是准确的,故而不必设置单个系统的独立帧刷新率。

## 2.5 先缓存式干涉检查

参数化轨迹法使用了不同于离散时间模型的干涉检查方法,由于所有对象的运动定义成参数形式,干涉检查操作可以通过比较一组路径观察它们的边界盒是否在某个公差范围内的方法实施。如果确实在一个公差范围内,可以计算它们在某个特定时刻路径是否相交。如果答案是肯定的,则出现了干涉碰撞现象,这时须详细计算两个对象的准确的干涉情况,并计算修改的路径。

与前述的离散时间干涉检查的逐帧计算机制不同,这种方法可以根据对象的运动路径在碰撞出现以前就计算碰撞,所以省去这些试验环节。另外,由于碰撞计算可以在碰撞以前完成,这可以减轻或避过处理平台的高峰计算压力。一旦得到碰撞计算的结果,计算机又转去运算新的路径,这样,计算机就有能力完成在新的路径可能出现的其他碰撞。当然这还需要建立一个修改堆栈系统。

## 2.6 固定的时间补偿

现有的虚拟现实系统可分为模型管理、通讯和渲染三个阶段。用户可以明显看到的图象移动滞后,除非模型管理和最后结果显示阶段的计算量特别小。移动滞后的程度随工作站加载速度的不同而不同。如果能对滞后量作出估算,那么可以在时间代码中加上这部分滞后时间,这样,观察者就能看到和真实情况更一致的图象。换句话说,针对不同系统给予不同的固定时间补偿,移动滞后可以部分或全部消除。

## 2.7 用户运动的表示

前面已经解决了对象运动路径的表示问题,用户和对象的交互问题也是必须考虑的。用户活动的信息是由跟踪系统捕捉的,在虚拟世界中控制一组用户物体,这组用户物体代表用户的身体。因此,如果跟踪系统频带宽度允许的话,用户运动在比较平滑阶段能够可以采用参数曲线拟合或卡尔曼方法转换成参数化形式,这样就可以预先评估用户的运动。

这种用户运动的策略有助于减少由于单个用户更改其他虚拟世界中的模型而给网络带来的广播信息。

## 2.8 长通讯延迟的效果

除了考虑单台机器的操作特性外,还必须考虑介入大量信息通讯延迟的问题。在现行系统中,延迟的介入严重影响了感知的平滑度和系统的响应。连续运动质量和状态传输性能都会因等待时间增加而退化。

而对于参数化系统,如果获得了足够的路径计划,用户可以看到模型运动是高度连续的。用户交互引起模型更改,等待时间将让远程系统直接跳到新的轨迹计划,然后与其他系统同步,如果路径定义合适的话,就可以减少跳跃。

如果考虑了物体与物体交互,则有两种可能发生的情况。如果考虑了碰撞类操作引起了主体的变化,而这些操作又都是本机上实施的,那么所有系统的模型都是一致的,导致网络的等待时间为零;如果干涉碰撞无法在本机确定,等待时间问题就类似用户与对象的交互。如果应用信息包改变了数据库,效果也与用户与对象交互的结果一致。

## 3 结论

本文介绍了一种实现共享和维护多用户虚拟现实系统的新方法,它同现有的虚拟现实系统相比有很多好处。首先,模型运动的同步校正可以在广域网络,甚至在 Internet 上进行;第二,其在广域网络上解决长时间延迟的能力较强;第三,在虚拟世界上对象运动时,可以大幅度减少网络上信息量;第四,对于变化的帧刷新率,具有可视性修正模型的能力;第五,减少或消除显示移动滞后问题;另外还具有增加捕捉用户动作的平滑度的能力。系统的缺点是要求有一个同步时钟服务器以及增加了操作系统的复杂性。

## 参考文献:

- [1] Sethi I K, Salari V, Vemuri S · Feature Point Matching Using Temporal Smoothness in Velocity, Pattern Recognition Theory and Applications [A]. NATO ASI Series, Vol. 30, Springer Verlag, Berlin, Germany, 1987.
- [2] Warwidd K, Rees D · Recursive Methods in Identification, Signal Processing for Control [M]. London: Peter Pepergrinus Ltd, 1986.
- [3] Roberts D J, Griffin M P · Distributed Virtual Reality Systems [A]. SERC/DTI Symposium [C]. Coventry School of Art, Coventry, UK, 1992.
- [4] Grimsdale C · Distributed Architectures for Virtual Reality [A]. IEE, Colloquium Digest No 1993/121, London, UK, 1993.
- [5] Comer D E · Interworking with TCP/IP [M]. London: Prentice-Hall Publications, 1991.

**作者简介:** 屠立忠, 博士研究生, 香港理工大学制造工程系助理研究员, 南京航空航天大学计算中心讲师; 季\_\_平, 博士, 香港理工大学制造工程系助理教授; 左敦稳, 博士, 南京航空航天大学机电工程学院教授; 王\_\_珉, 博士, 南京航空航天大学机电工程学院教授, 博士生导师。