渲染机群管理系统负载平衡算法的研究与实现

傅 游,李丽丽,花 嵘
(山东科技大学 信息科学与工程学院,山东 青岛 266510)

摘 要:机群渲染系统负载平衡是为了及时地解决在动漫渲染制作中出现的不平衡问题,使得机群渲染系统能够充分地利用好每个工作结点,以此来提高渲染机群管理系统的工作效率,解决动漫制作中渲染这一瓶颈。设计了动漫渲染管理系统的软件平台架构,并详细说明了管理系统的工作流程。对其中的关键部件的功能和结构进行了设计,具体给出了一个负载均衡算法,优化了渲染系统中的负载平衡算法。最终的测试结果表明了算法的有效性。

关键词:动漫渲染;机群系统;软件架构;负载平衡;机群管理系统
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1673-629X(2011)07-0094-04

Research and Implementation of Load Balance Algorithm of Animation Rendering Cluster Management System

FU You, LI Li-li, HUA Rong
(College of Information Science and Engineering. Shandong University Science and Technology, Qingdao 266510, China)

Abstract: Load balance of cluster rendering system is for resolving the unbalance problem timely in the making of animation rendering to makes the working nodes can render fully and make cluster rendering system can fully utilize. In order to improve the working efficiency, so as to solve animation manufacture the problem of bottleneck in rendering, designed a cartoon rendering management system software platform structure, and detail the management work flow of the system. And for the key components of function and structure design, and specifically given a load equalization algorithm and optimized the rendering system of load balancing algorithm, the final test results show that the effectiveness of the proposed algorithm.

Key words: animation render; cluster system; software structure; load balance; cluster management system

0 引 言

渲染[1]是指在计算机内建立的3D几何模型上附加一定的材质、纹理及色彩,并加上光源,通过计算机的计算生成具有真实感效果的场景图形。随着现代动漫制作中渲染的计算量不断增大,渲染计算时间越来越长,渲染已成为动漫产业发展的最大瓶颈[2]。随着计算机网络技术和中间件技术的发展,机群系统[3]的应用领域不断扩张,将渲染技术与机群系统结合的渲染机群系统成为动漫渲染的主流。

中国目前有国家级动漫基地52个,均采用渲染机群为动漫企业提供渲染平台。如湖南国家数字媒体技术产业化基地三维机群渲染系统拥有100个CPU;上海市多媒体公共服务平台使用了清华同方的基于GPU的机群渲染系统;广州影视动画渲染中心拥有1000个CPU。其中大部分渲染机群系统是通过简单的自动分配来完成指定的渲染工作,导致渲染工作效率低、负载不均衡、容错能力差等问题。文中设计了机群渲染系统的管理模型,给出了渲染机群系统负载平衡的算法,测试结果说明该系统能够提高渲染的效率,缩短渲染时间。

1 渲染机群管理系统及软件架构

渲染机群(Render Farm)[4]是指利用机群计算系统的优势,通过网络分发软件和并行渲染软件,充分利用机群系统中的计算机硬件资源,将复杂的场景通过大量的并行计算,生成预览图像或者最终动漫图像,以供效果调整审定和后期制作合成之用[2]。

1.1 渲染机群管理平台软件架构

渲染机群管理系统[5]就是根据用户的渲染需求来统一协调和管理机群的软硬件资源,保证用户渲染作业能够公平合理地共享机群资源,提高系统利用率和

收稿日期:2010-12-08;**修回日期**:2011-03-10
基金项目:山东省教育科技计划项目(J08LJ11);青岛经济技术开发区科技发展计划项目(2008-02-27)
作者简介:傅 游(1968-),女,教授,研究方向为网格计算及分布式系统。

吞吐率,最终快速、高效地完成用户的渲染要求。一般渲染机群管理平台[6]是由客户端、管理节点、计算节点以及共享文件系统构成,在本系统设计中采用了 master-slave 的架构模式,使用的是 MPI[7] 的运算模式,它的软件架构如图 1 所示。

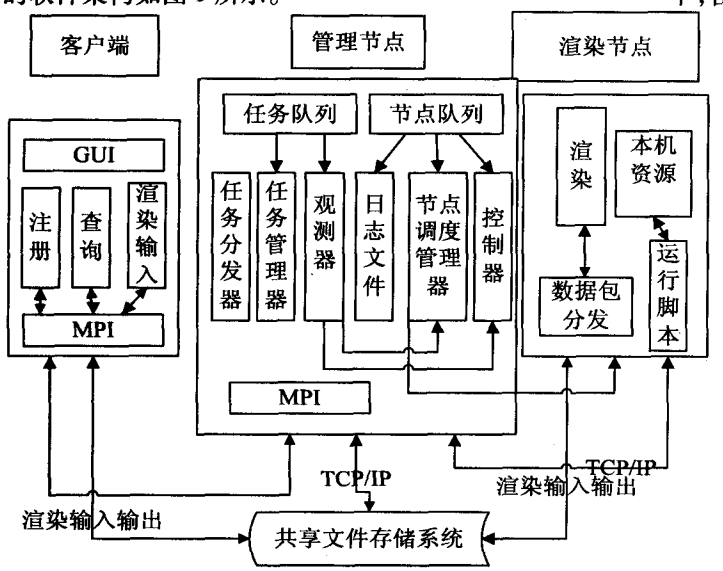

图 1　渲染机群管理平台软件架构

1.2　渲染机群管理系统的工作流程

在该软件架构下渲染机群管理系统的工作流程为:首先客户端利用 3D 软件制作出 3D 场景,将场景文件输入到共享文件存储系统中;然后通过管理软件接口将该文件提交给渲染机群管理节点;接着启动渲染管理软件,通过管理节点查找网络上的空闲节点,通过任务分发器把渲染任务分配到空闲渲染节点上;空闲渲染节点接到任务后,利用自己的渲染引擎开始渲染工作,并把工作情况实时或定时报告给渲染管理节点;渲染管理节点把收到的负载、资源利用率等信息反馈给管理软件,同时按用户需求进行信息反馈;整个场景渲染完成后,将结果输出再输出到文件服务器中,用户得到可以应用的图片序列。

1.3　主要部件结构与功能

该管理平台中观测器采集渲染机群系统资源(机器资源信息和渲染任务信息)信息,并将原始数据记录在日志文件中,控制器接收来自观测器[8]的信息,利用这些信息对机群系统资源使用情况和发展趋势进行预测[9],并根据预测的结果进行负载均衡。

渲染机群观测器主要由监测器、日志文件、信息分析器、信息预处理器、信息聚合器几个模块组成(见图 2)。监测器负责周期性地采集渲染集群系统的资源状况信息,为控制器的工作提供最原始渲染集群系统资源样本,并且将收集到的原始数据记录在日志文件中,在系统资源预测分析过程中使用。信息预处理器对系统资源原始数据进行预处理,得到统一的数据形式,同时将这些数据传送到信息聚合器中。信息分析器分析整个系统的实时信息,对信息进行规范化处理,提供系统资源预测描述信息,同时将这些信息传送到信息聚合器中。此时,信息聚合器将这些数值和原始历史记录存储到本身的存储器中,存储器中的这些信息都需要执行筛选过程,过滤掉影响预测结果的不利因素。同时聚合器会将筛选好的信息和历史信息一起传送给控制器,这些信息都会被抽象成对当前系统资源状态和变化趋势的描述。

控制器首先接收来自观测器的信息,动作选择器根据预测结果调用调度算法,产生最有利的诱导动作,并将该诱导动作传递到机群渲染系统中。架构中的自适应模块利用遗传运算或者进化算法生成新的规则,引入新的诱导动作,并将诱导动作的执行结果传送到评估模块,评估模块根据执行结果判断该诱导动作是否符合用户要求。

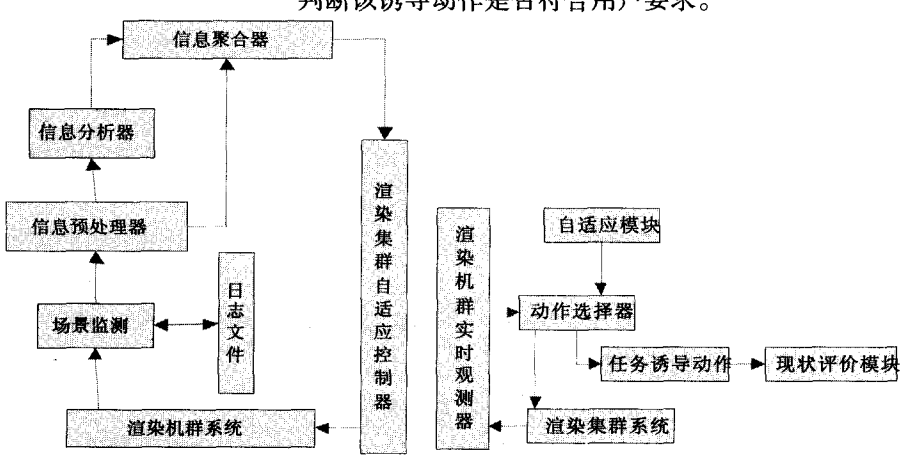

图 2　机群渲染系统观测器架构与机群渲染系统控制器架构

整个渲染机群管理系统的通过观测器和控制器等部件实现了系统的信息采集、分析及负载平衡功能。由于篇幅所限,文中主要讨论机群渲染中的负载平衡算法[10]。

2　渲染机群系统负载平衡算法的设计实现

常见的调度[11]算法主要有 FCFS、SPT、LSF、RM 等。其中,FCFS 最大的优点是易于高效实现,而且能够保证系统的公平性,作业的执行顺序是可以预见的。

但是它牺牲了系统的吞吐率和利用率,并且会造成系统资源的浪费。短时间作业优先(SPT)算法提高系统的作业吞吐能力,降低作业的平均等待时间,会导致那些执行时间长但是资源容易得到满足的作业等待时间太长。LSF(最小空闲时间优先算法)能够按照任务被调用的缓急程度对任务进行调度,但是它增加了系统的开销,造成了任务之间的频繁切换和严重的颠簸现象。RM算法根据任务的周期指定任务的优先级,周期越短,优先级越高。算法总是试图调度优先级高的任务级高的任务,这样会使得那些迫切需要但是周期长的任务得到不到调度。

文中结合速率单调算法(RM)、多参数调度策略与最小空闲时间优先算法(LSF)的优势,给出一种动态参数调度策略。从调度策略中调度优先级确认和动态生成抢占阈值的方法出发,最终给出动态调度策略的算法。

2.1 调度优先级的确认

文中任务的优先级用一个三元组 $D(t, s, j)$ 来表示,其中 j 表示任务的优先级,s 表示任务的空闲时间,t 表示任务的执行周期。任务执行周期与空闲时间越少,任务的优先级就越大。

这里引入了优先级权重,假设RM算法优先级权重为 $X(t,s)$,LSF算法的优先级权重 $Y(t,s)$,那么动态调度策略优先级为:

$$j = \alpha * X(t, s) + \beta * Y(t, s) \quad (1)$$

其中 $\alpha + \beta = 1$,在初始情况下,α 与 β 均为0.5,可以根据具体的任务情况来调整 α 与 β 的取值。如果要求执行周期短的任务尽快执行,就将 α 参数的值调大,β 参数的值调小;如果要求紧急任务尽快执行,就将 β 参数的值调大,α 参数的值调小。

2.2 抢占阈值动态生成

为了防止系统因为上下文切换,产生抖动现象,引进了动态生成抢占阈值思想[10]。

对于抢占阈值的计算给出一个线性方案:

$$h = \gamma * j(0) + (M - \gamma * j(0)) * (j - j(0))/(M - j(0)) \quad (2)$$

M 为优先级的最大值;γ 为一个系数常量,$\gamma > 1$;$j(0)$ 表示任务的初始优先级。

2.3 动态调度策略调度算法的实现

文中设计的算法中创建了两个链表,分别用来表示任务周期与空闲时间,为了便于对优先级权重进行标识对这两个链表进行降序排序。任务周期越小,它在链表中的位置就越靠后,它的下标值就越大。同理,空闲时间越小,它在链表中的位置就越靠后,它的下标值就越大。所以,可以用任务周期链表中任务的下标值来表示权重 $X(t,s)$;用空闲时间链表中任务的下标值该表示权重 $Y(t,s)$。整个算法如下:

```
Task = Class
Private
  s:  integer;  //空闲时间
  t:  integer;  //任务周期
  j:  integer;  //任务优先级
  h:  integer;  //任务抢占阈值
  c:  integer;  //预留资源指,本算法中表示任务所需的CPU数
  RM_b,RM_E:  pRMlst;  //任务周期链表头尾指针
  LSF_b,LSF_E:  pLSFlst;  //空闲时间链表头尾指针
End;
while(遍历指针 p <=任务周期链表尾指针 RM F_E) do //提取RM权重
begin
if 新任务空闲时间 > p.s  then  //新任务的周期与链表中任务的周期比较来确定
    break;              //新任务在链表中的位置
  p := p.Next;
  end;
while(遍历指针 p <=空闲时间链表尾指针 LSF_E) do  //提取LSF权重
begin
if 新任务空闲时间 > p.s then  //新任务的空闲时间与链表中任务的空闲时间比较
    break;              //用来确定新任务在链表中的位置
  p := p.Next;
  end;
```

获得RM与LSF的权重后,利用式(1)计算出任务的优先级,同时根据优先级利用式(2)计算出抢占阈值。

将计算出的优先级与系统中正在执行任务的抢占阈值进行比较,如果任务的优先级低于正在执行任务的抢占阈值,则不进行调度。如果任务的优先级高于正在执行任务的抢占阈值,此时检查当前系统空闲的CPU数是否满足任务所需的CPU数目,如果满足,就立刻进行调度,如果不满足,就等到CPU资源满足任务的需求时,才进行调度。

3 渲染机群管理系统的部署与测试

为了验证文中的负载平衡[12]算法的效果,在局域网内采用Deadline渲染管理软件将6台机器构成渲染机群系统,进行了测试。机器配置:CPU为AMD Opteron 2.19 GHz,2GB的内存。测试中采用以前的没有负载均衡功能的渲染机群管理系统和文中设计的带有负载均衡功能的渲染机群管理系统,对80帧的文件进行渲染。分别在第10、20、30…70帧时记录CPU、内存的利用率以及任务的平均等待时间等性能指标,得

到两个系统的渲染任务等待、完成时间对比图(如图 3 所示)和系统节点的内存、CPU 利用率对比图(如图 4 所示)。

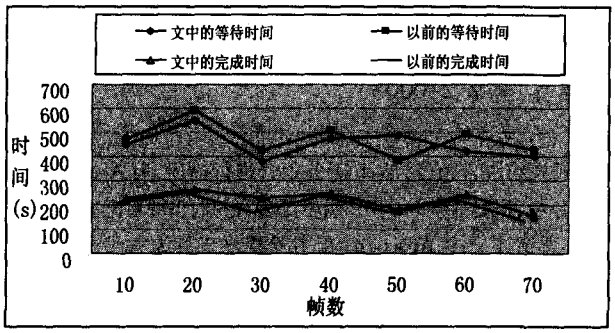

图 3　两个渲染机群系统的渲染任务等待、完成时间对比图

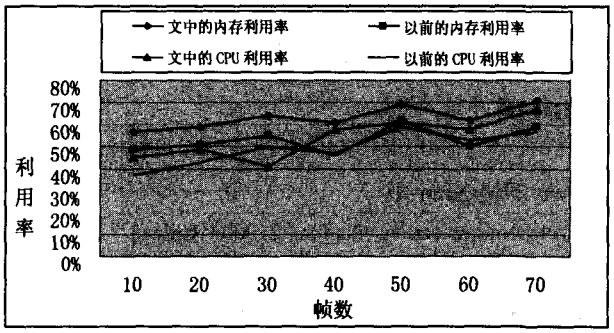

图 4　两个渲染机群系统节点的内存、CPU 利用率对比图

通过实验结果分析可以看出,文中设计的渲染机群系统在 CPU 与内存的利用率上有了明显提高,任务的等待时间也明显减少。

4　结束语

文中需要进一步研究的内容是改进动态参数调度算法,设计出更为详细的参数影响系数,还需要对渲染集群系统进行功能的扩展,如设计安全性服务,以保证用户和数据的安全性。

参考文献:

[1] 王　钰. 基于有机计算的动漫渲染集群系统管理技术的研究与应用[D]. 青岛:山东科技大学,2010.

[2] 陈　颉,周智等,黄刘生. 负载平衡在三维渲染中的应用[J]. 计算机工程与应用,2005(34):65-67.

[3] 罗秋明,孙宏元. 集群渲染管理软件的构建技术与框架设计[J]. 计算机工程, 2008,34(11):249-251.

[4] Madhavan K P C, Arns L L, Bertoline G R. A Distributed Rendering Environment for Teaching Animation and Scientific Visualization[J]. Computer Graphics and Applications, 2005, 25(5):32-38.

[5] 龚　雷,李　晨. 基于 3DSMAX 二次开发的批量渲染系统[J]. 计算机技术与发展,2009,19(2):102-105.

[6] Chiang Chuanwen, Lee Chungnan, Chang Mingjyh. A dynamic grouping scheduling for heterogeneous Internet-centric meta-computing system[C]//Proceedings of 8th International Conference on Parallel and Distributed System. [s. l.]: Institute of Electrical and Electronics Engineers Computer Society, 2001:77-82.

[7] 卢　照,张锦娟. MPI 动态负载平衡策略的研究与实现[J]. 计算机技术与发展,2010,20(5):132-135.

[8] Richter U, Mnif M, Branke J, et al. Towards a generic observer/controller architecture for organic computing[C]//In Hochberger C, Liskowsky R. INFORMATIK 2006 - Informatik fur Menschen. Volume P-93 of GI-Edition - Lecture Notes in Informatics (LNI). [s. l.]: Bonner Kollen Verlag, 2006:112-119.

[9] 金　宏,王宏安,王　强,等. 改进的最小空闲时间优先调度算法[J]. 软件学报,2004,15(8): 1-8.

[10] 季　华,陈福民. 带负载平衡策略的 PCsort-first 并行渲染系统[J]. 计算机仿真,2005,22(11):209-214.

[11] 徐慧慧,石　磊. 网格资源调度算法研究[J]. 计算机技术与发展,2009,19(9):76-80.

[12] 徐　群,祝永志. 集群系统中的负载均衡问题的研究[J]. 计算机技术与发展,2009,19(8):129-134.

(上接第 93 页)

ACM, 2000, 43(8):40-43.

[3] Borst W N. Construction of Engineering Ontologies for Knowledge Sharing and Reuse [M]. Enschede: University of Twente, 1997:68-70.

[4] 曹泽文,李立人,邓　苏. 战场空间本体构建方法研究[J]. 火力与指挥控制,2008(6):22-25.

[5] 李军勇,宋俊峰,阳东升. 面向网络中心战的规则本体构建研究[J]. 舰船电子工程,2008,28(21):1-4.

[6] 张东辰, 周　吉. 军事通信[M]. 第 2 版. 北京:国防工业出版社,2008:173-212.

[7] Falconer S M, Margaret-Anne Storey. A Cognitive Support Framework for Ontology Mapping[C]// 6th International and 2nd Asian Semantic Web Conference. [s. l.]:[s. n.],2007:114-127.

[8] OWL Web Ontology Language Guide[EB/OL]. 2004-01. http://www.w3.org/TR/owl-guide.

[9] Horridge M. A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools Edition 1.2[M]. University of Manchester, 2009:70-75.

[10] 黄珂萍, 蒋昌俊. 基于本体的城市交通的知识分析和推理[J]. 计算机科学,2007,34(3):192-196.

[11] 王继东,张　瑜,李　娜. 基于本体的语义检索技术研究与实现[J]. 计算机技术与发展,2009,19(10):134-137.

[12] 王金环, 李宝敏. 基于本体 DL 的语义推理研究[J]. 计算机技术与发展,2009,19(11):94-96.