第33卷 第18期 计 算 机 工 程 2007年9月
Vol.33 No.18 Computer Engineering September 2007
·软件技术与数据库· 文章编号：1000—3428(2007)18—0045—03 文献标识码：A 中图分类号：TP31

可扩展并行作业调度模拟器 ParaSim 设计与应用

罗红兵，张 宇，张晓霞
(北京应用物理与计算数学研究所高性能计算中心，北京 100088)

摘 要：针对作业调度研究的需求，设计和实现了一个可扩展的并行作业调度模拟器 ParaSim。ParaSim 采用与实际并行作业系统近似的工作流程，以资源占用矩阵来表示计算资源，使用事件驱动的模式进行模拟调度和运行，支持空间共享和时间共享等多种调度策略，并允许对各调度参数进行设置。ParaSim 已投入实际使用，为并行机作业调度策略的定量分析、调整和优化提供了有力的支持。
关键词：大规模并行计算机；作业调度；调度评价；模拟器

Design and Application of Scalable Parallel Job Scheduling Simulator ParaSim

LUO Hong-bing, ZHANG Yu, ZHANG Xiao-xia
(High Performance Computing Center, Institute of Applied Physics and Computational Mathematics, Beijing 100088)

【Abstract】 This paper introduces a scalable parallel job simulator named ParaSim. ParaSim makes use of the event-driven executing model and adopts matrix to represent computation resources, so it can support both space-sharing and time-sharing scheduling strategies in the same framework. It also provides many policies and scheduling parameters like real system. At present, ParaSim has been implemented and used in scheduling strategy research. The experiment based on ParaSim shows that it can benefit from ParaSim on research and selection of scheduling strategies in real job system.
【Key words】 massive parallel computer; job scheduling; evaluation of scheduling; simulator

1 概述

随着越来越多的大规模并行计算机和集群系统投入使用，并行作业的调度问题越来越受到关注，其核心是调度策略和算法的研究，涉及作业流、用户约束、调度算法、回填策略等多个因素。研究表明[1]不存在理想的调度策略适用于各种并行编程语言、体系结构和操作系统，且作业流特征对调度策略有重要影响[2]。并行作业调度研究的重点在于针对具体的并行机和作业流研究合理的调度策略。由于并行计算机是稀缺资源且往往处于生产性运行中，故完全以实际并行机作为实验平台并不合适。可行的方法[5]是建立并行调度模拟器，在模拟器上进行前期的算法研究和对比，然后再到实际并行机上进行后期验证，从而设定合理的调度策略和算法。

尽管使用模拟器是进行调度研究的常见方式，但目前的作业调度模拟器[3~5]存在以下不足：(1)缺乏对作业调度策略的完整模拟，实际的作业调度涉及调度策略和算法、资源使用策略、用户策略等多因素，而现有的模拟器多局限于调度算法的模拟；(2)作业调度依赖于并行计算机体系结构，并受调度开销、资源信息获取延迟等的影响，而现有的模拟器局限于特定体系结构，可扩展性受限，且一般都忽略了上述因素的影响。

针对现有作业调度模拟器的上述不足及实际算法研究的需要，笔者着手进行了并行作业调度模拟器(ParaSim)的设计与实现，期望利用该模拟器对并行作业的调度和执行过程进行接近真实的模拟，同时兼顾可扩展性。目前 ParaSim 主要用于 MPP 系统和集群系统上的作业调度模拟，其特点如下。

(1)可扩展性：ParaSim 中采用了事件驱动的模型，保持了作业队列、调度策略、模拟执行、资源等关联实体间的独立性，使模拟器具有较好的可扩展性；

(2)灵活性：ParaSim 允许设置调度策略、回填策略、用户进程数限制等，并支持新调度算法的添加，可以获得各策略在不同组合下的调度效果；

(3)真实性：ParaSim 允许用户进行调度开销、加载开销、资源信息获取延迟等参数设置，与实际调度结果的误差较小；

(4)便利性：ParaSim 提供公平性和效率等多评价指标的计算且允许定制，方便了对调度策略的评价。

2 基本工作流程

ParaSim 采用与实际并行作业调度类似的工作流程(如图1)。作业日志作为 ParaSim 的输入，它通过作业流生成器处理形成模拟调度所需作业流；作业流按时间顺序依次进入等待队列；ParaSim 中的作业调度器则依照调度策略和资源状况，选择作业放入运行队列；模拟执行器负责运行队列的模拟执行，当作业的累计执行时间达到实际执行时间后，结束该作业的执行；当作业流中的所有作业执行完毕后，一次模拟调度过程结束，此时统计程序计算出相应的评价指标。

ParaSim 的核心是作业调度器和模拟执行器，其模拟过程采用事件驱动的模式，4 个主要事件为：(1)到达事件，在作业第 1 次提交并放在等待队列中时发生；(2)调度事件，在作业分配到模拟环境中时发生，作业调度器周期性检查到达事件和完成事件的发生，当等待队列存在作业且满足资源要求，则发生调度事件；(3)开始事件，调度事件发生，延迟一

基金项目：中国工程物理研究院基金资助项目(20060646)
作者简介：罗红兵(1968－)，男，副研究员，主研方向：高性能计算和网格计算；张 宇，助理工程师；张晓霞，高级工程师
收稿日期：2006-09-30 **E-mail：**hbluo@iapcm.ac.cn

定时间(即用户设定的加载开销)，发生开始事件；(4)完成事件，当作业完成时发生，此时释放资源。

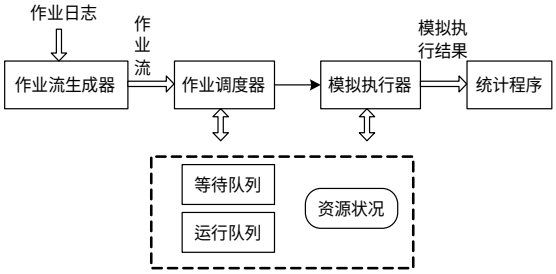

图 1　ParaSim 结构示意图

3　资源表示和使用

资源表示是模拟器设计的重要环节，关系到模拟调度和执行的设计和实现细节。考虑到 MPP 和集群系统中各计算结点间的通信开销基本一致，ParaSim 的资源模型是一个以 CPU 资源和内存资源为核心的表示，不考虑 CPU 间的通信关系。

由于目前作业系统中常见的调度策略主要分为空间共享模式和时间共享模式两类，ParaSim 采用资源占用矩阵来表示模拟器中的虚拟机资源。矩阵的行表示 CPU 资源的一个时间片，列表示某一 CPU，每个单元表示某一 CPU 的一个时间片。时间共享的度用多道程序数(MPL)来表示，当 MPL 数为 1 时，就表示使用空间共享模式。另外为实现上的便利，ParaSim 假定时间片是固定大小，且任务分配是静态的，不能迁移。

图 2 是一个 8 处理器模拟机(MPL为 4)的资源占用矩阵示意图，图中 J_i^j 表示资源被分配给作业*i*的第*j*个任务。与实际系统类似，ParaSim中使用资源的原则是尽可能调度更多的作业，当作业分配完成后，再扩充作业使之占据矩阵中的多行，尽可能地利用计算资源，如图 2 中作业J_4和J_5都使用了 2 个时间片。

	P_0	P_1	P_2	P_3	P_4	P_5	P_6	P_7
时间片 1	J_1^0	J_1^1	J_1^2	J_1^3	J_1^4	J_1^5	J_1^6	J_1^7
时间片 2	J_2^0	J_2^1	J_2^2	J_2^3	J_2^4	J_2^5	J_2^6	J_2^7
时间片 3	J_3^0	J_3^1	J_3^2	J_3^3	J_4^0	J_4^1	J_5^0	J_5^1
时间片 4	J_6^0	J_6^1	J_6^2	J_6^3	J_4^0	J_4^1	J_5^0	J_5^1

图 2　资源占用矩阵示例

4　模拟调度方法

资源占用矩阵是 ParaSim 的核心，作业调度模拟围绕着资源占用矩阵进行。作业调度器会周期性检查到达事件和完成事件的发生，当有这些事件发生时，它按如下步骤进行作业调度和资源分配。

步骤 1 资源整理

资源整理的目的是调整资源占用矩阵，使系统能尽可能调度更多的作业，但资源整理仅在 MPL 大于 1 时有效。资源整理过程为：首先去除作业在资源占用矩阵中占据的多行，仅保留其中一行，以提供更多的机会给等待作业；然后是重新组织矩阵，类似于碎片处理，把作业从较空的行移动到较密的行，从而提供更空的大时间片以利用大作业调度；重新组织矩阵的过程是从任务数最少的行开始到任务数最多的行，遍历矩阵，为每行的作业找新目标行，目的是把最多的作业聚集在最少的行中。由于 ParaSim 中任务分配是静态的，作业移动时必须是对应的目的列为空，以保证任务所在的处理机不变。

步骤 2 作业调度

根据调度算法在等待队列中选择作业，直至无法满足等待作业的资源要求。调度算法关系到作业和资源的选择，由用户定制，目前 ParaSim 包括的调度策略有：先到先服务(FCFS)，大作业优先，短作业优先，长作业优先和 FirstFit 等。ParaSim 中作业调度还考虑了系统对每个用户允许的最大任务数的约束和回填策略，其基本过程如下：

```
IF    (user policyexists)
Filt    queue；//过滤掉不符合模拟要求的作业
Waiting queue sort ;//根据调度算法的要求进行排序,或者不排序
WHILE (waiting queue is not empty){
  get head of queue( job j );
IF (enough resources are free to start j ) {
        start j and Fill Resource Matrix;
        remove j from queue;
}
  ELSE IF (backfill policy exists)//如果使用回填
     backfill other jobs from the queues; //根据回填策略
}
```

步骤 3 资源矩阵扩充

按一定的策略扩充作业，使之占据多行，填充矩阵中的空洞。资源矩阵扩充也有多种策略，目前 ParaSim 是按作业调度策略来设定资源矩阵扩充策略。资源矩阵扩充遵循任务分配静态性原则，任务的列不能改变。该步骤同样在 MPL 大于 1 时有效。

5　性能评价指标

ParaSim参考了国外作业调度策略评价指标[3~5]，选择了平均响应时间、平均等待时间及系统利用率作为作业调度性能评价指标。此外考虑到平均响应时间和平均等待时间的计算未对作业的类型和规模加以区分，ParaSim还引入加权平均响应时间和加权平均等待时间来进一步对作业系统性能和作业流特点进行衡量，权值即作业并行度的体现。

具体而言，平均响应时间(ART)、平均等待时间(AWT)和系统利用率(Efficiency)计算公式如下：

$$ART = \frac{1}{N}\sum_{j\in Jobs}(\mathrm{EndTime}_j - \mathrm{submitTime}_j) \tag{1}$$

$$AWT = \frac{1}{N}\sum_{j\in Jobs}(\mathrm{StartTime}_j - \mathrm{submitTime}_j) \tag{2}$$

$$Efficiency = \frac{\sum_{j\in Jobs}(EndTime_j - StartTime_j)\times CPU_j}{(EndTime_{last\_job} - SubmitTime_{first\_job})\times TotalCPU}\times 100\% \tag{3}$$

其中，$\mathrm{SubmitTime}_j$、$\mathrm{StartTime}_j$、$\mathrm{EndTime}_j$分别表示作业j提交给队列、开始运行和完成时间。

加权平均响应时间(PART)和加权平均等待时间(PAWT)的计算公式如下：

$$PART = \frac{\sum_{j\in Jobs}(\mathrm{EndTime}_j - \mathrm{submitTime}_j)\times CPU_j}{\sum_{j\in Jobs} CPU_j} \tag{4}$$

$$PAWT = \frac{\sum_{j\in Jobs}(\mathrm{StartTime}_j - \mathrm{submitTime}_j)\times CPU_j}{\sum_{j\in Jobs} CPU_j} \tag{5}$$

6　模拟器的应用

笔者已用 Java 语言编程实现了 ParaSim 模拟器，并利用实际作业日志进行了如下一些实验和研究。

6.1　作业流分析

为准确了解并行机系统的运行状况，需要对作业日志进行统计分析，利用 ParaSim，就可以方便地完成该项工作，如表 1 和表 2 就是通过 ParaSim 获得的两个时段作业按任务

数的分布情况，它们的基本特征如下：

(1)作业流 1：总作业数：4 103，平均执行时间：130 422s，平均并行度：41.8。

(2)作业流 2：总作业数：3 250，平均执行时间：6 408s，平均并行度：61.2。

表 1　作业流 1 按任务数的分布

任务数	数量	所占比例/%
1	2 551	62.2
2	4	0.1
≤4	56	1.4
≤8	77	1.9
≤16	100	2.4
≤32	457	11.1
≤64	146	3.6
≤128	347	8.5
>128	365	8.9

表 2　作业流 2 按任务数的分布

任务数	数量	所占比例/%
1	979	30.1
2	86	2.6
≤4	87	2.7
≤8	120	3.7
≤16	468	14.4
≤32	432	13.3
≤64	377	11.6
≤128	240	7.4
>128	461	14.1

6.2　调度过程分析

为优化调度算法，对调度过程进行详细分析是十分必要的。ParaSim 可以记录调度过程中的执行细节，帮助研究人员从中发现调度算法的不合理之处。图 3 就是使用结合回填的 FCFS 策略在 CPU 数为 640 的模拟机上执行作业流 2 时某时段的细节，记录的信息是作业系统调度作业前在用 CPU 资源情况(三角形点线)和同时刻所有等待作业的资源请求(圆形点线)。

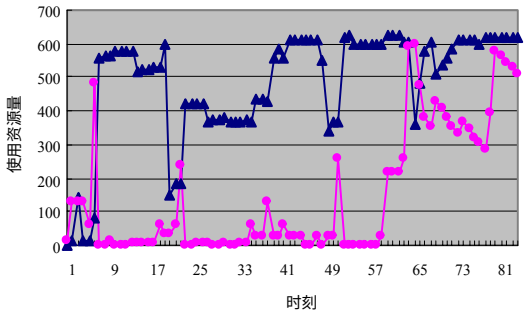

图 3　系统资源状况和资源请求状态

从图 3 中可以发现以下两种现象：(1)有时系统空闲资源较少，但系统中资源请求总量较大且单个请求的资源量多，例如在图 3 中的时间点 67~83，被使用的 CPU 数量已经达到 500~610，但等待作业的总 CPU 请求数达到了 300。很明显此时受 FCFS 的次序约束和回填要求的限制，会造成部分资源无法被利用，影响系统利用率和作业整体响应时间。(2)有时系统空闲资源多，但资源请求总量和单个请求的资源量都较少，例如从时间点 20 至 39，超过 200 个 CPU 处于空闲状态，但等待队列中总 CPU 需求低于 100，显然这个时段系统利用率也是低下的。上述调度过程分析显示：作业调度过程中的资源和用户请求不匹配是影响到系统利用率的问题之一，改进资源与请求间的匹配度，是改进调度效果途径之一。

6.3　调度算法效果分析

为在实际系统中选择合理的调度策略，就需要对比不同调度策略的调度效果。表 3 和表 4 是利用模拟器对 FirstFit、结合回填的 FCFS、短作业优先(SRTJF)、大作业优先(BJF)等空间共享策略的对比实验结果。表中结果数据显示，对工作流 1，不同策略差别很小，经分析发现这是由于该作业流中串行作业占 6 成且平均执行时间较长，回填作用不大而造成的。对工作流 2，不同策略调度效果的差别有所体现且回填有明显作用。除短运行时间作业优先策略效果最好之外，结合回填的 FCFS 和大作业优先策略的效果都优于 FirstFit。另外测试数据表明：用户对作业执行时间估计不准对回填的调度效果有影响，但影响不大。

表 3　不同调度策略的调度效果对比(作业流 1)

评价指标/算法	FirstFit	FCFS+B	BJF+B	SRTJF
ART	143 244	142 149	142 464	144 037
AWT	12 821	11 726	12 042	13 614
PART	101 407	102 607	96 822	133 460
PAWT	64 788	65 987	60 203	96 841

表 4　不同调度策略的调度效果对比(作业流 2)

评价指标/算法	FirstFit	FCFS+B	FCFS+B®	BJF+B	BJF+B®	SRTJF
ART	83 535	79 075	80 492	78 498	79 989	76 567
AWT	19 454	14 993	16 411	14 417	15 907	12 486
PART	90 794	88 258	87 315	78 932	79 863	79 830
PAWT	66 868	64 332	63 389	55 006	55 937	55 903

表中，®表示 4 次模拟执行结果的平均(每次用户估计的执行时间为实际执行时间的 1~2 倍；各时间的计量单位是秒。

7　结论和进一步的工作

ParaSim 的实现和投入使用给并行作业调度的研究和实际作业调度策略的选择和优化带来了很大的便利。目前，笔者尚未将内存和通信等在资源表示中实现，下一步将根据实际研究中的需要加以扩充完善。另外，针对真实作业流获取难的问题，笔者还准备将随机合成作业流的机制引入到 ParaSim 中。

参考文献

1 Feitelson D G. The Forgotten Factor: Facts on Performance Evaluation and Its Dependence on Workloads[C]//Proc. of Euro-Par 2002 Parallel Processing. 2002: 49-60.

2 Feitelson D G, Rudolph L. Metrics and Benchmarking for Parallel Job Scheduling[C]//Proc. of the 4th Workshop on Job Scheduling Strategies for Parallel Processing, Orlando. 1998: 1-24.

3 Franke H, Jann J, Moreira J. An Evaluation of Parallel Job Scheduling for ASCI Blue-pacic[C]//Proceedings of SC'99, Portland, Oregon. 1999: 11-18.

4 Krevat E, Castanos J, Moreira J. Job Scheduling for the Blue Gene/L System[C]//Proc. of the 8th Workshop on Job Scheduling Strategies for Parallel Processing, Edinburgh. 2002: 38-54.

5 Kleban S D, Clearwate S H. Simulating Performance Sensitivity of Supercomputer Job Parameters[C]//Proc. of High Performance Computing Symposium. 2003.