

GPU 集群能耗优化控制模型研究

王海峰^{1,2}, 曹云鹏^{1,2}

(1. 临沂大学信息学院, 山东临沂 276005; 2. 山东省网络环境智能计算技术重点实验室临沂大学研究所, 山东临沂 276005)

摘要: 随着大数据技术的发展, GPU 集群作为一种高效的并行系统被应用到大规模数据实时计算中. 能量是实时计算时重要的资源, GPU 集群的能耗优化及实时消减成为一个具有挑战性的问题. 从集群全局角度引入模型预测控制策略, 并建立闭环反馈机制的多输入多输出控制器. 通过调整计算频率和改变活跃流多处理器来改变能耗状态, 利用反馈和滚动优化机制完成对未来的控制预判, 实现消减冗余能耗的目标. 实验表明: 控制模型的精度和节能效果优于基准模型, 而且具有较好的稳定性, 适合应用到大规模数据实时计算中.

关键词: 能耗控制; GPU 集群; 能量消减; 模型预测

中图分类号: TP302.7 **文献标识码:** A **文章编号:** 0372-2112 (2015)10-1904-07

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2015.10.004

Power Consumption Optimization Control Model of GPU Clusters

WANG Hai-feng^{1,2}, CAO Yun-peng^{1,2}

(1. Information School, LinYi University, Linyi, Shandong 276005, China;

2. Institute of Linyi University of Shandong Provincial Key Laboratory of Network based Intelligent Computing, Linyi, Shandong 276005, China)

Abstract: With the development of Big Data technology GPU cluster as a high efficiency parallel system applies into the Large-scale data computing field. Energy is a significant computation resource. So power consumption optimization control and capping in real-time becomes a challenge issue. The Model Prediction Control strategy is introduced and a Multi-Input Multi-Output controller is built by using a closed loop feedback principle from the whole cluster perspective. Power consumption status is changed by scaling frequency and adjusting active stream multi-processors. Then the feedback and the periodic optimization mechanisms can predict the control behaviors in the future control cycles. This achieves the goal that reduces redundancy energy. The results demonstrate that the proposed model has more accuracy and consumes less energy than the others. And it has better control stability. So it has better adaptability and obvious advantage in the Large-scale data real-time computing.

Key words: power consumption control; graphic processing unit (GPU) clusters; power capping; model prediction control

1 引言

随着云计算和物联网的发展, 大数据的时代已经来临. GPU (Graphics Processing Unit, GPU) 集群成为一种重要的大数据实时计算系统, 各类企业都拥有较大规模的计算集群^[1,2]. 在实时计算中 GPU 集群会产生巨大能耗, 因此 GPU 集群能耗优化和消减具有重要研究意义. 传统的 CPU 集群能耗优化是一个高性能计算与控制工程相结合的交叉研究方向. Horvath 将一个简单反馈控制应用到多层 Web 服务器集群的能耗优化控制中^[3]; Luciano 把反馈控制与混合整数规划相组合, 并构造出 Web 服务器集群的能耗控制模型^[4]. Wang 设计多输入

多输出能耗控制器, 在全局范围内设计集群各节点的协同优化控制模型^[5]. 然而, GPU 作为一种新兴的通用计算芯片, 与 CPU 具有明显不同之处, 不仅提高片上计算密度, 而且简化片内逻辑控制单元的设计. 因此 GPU 集群的能耗优化控制具有新的研究价值.

国内学者开始关注单节点任务调度的能耗分析与优化^[6,7], 以及能耗预测模型的研究^[8-10], 少量节能任务调度算法的研究可应用到 GPU 集群能耗优化中^[11]. 然而大规模数据的实时处理属于典型的多任务计算, 需要考虑计算任务的连续性和负载的动态变化. 特别是针对 GPU 能耗特点来建立 GPU 集群能耗控制模型, 在实时计算过程中指导能量配给管理是一个需要解决的科

学问题.为研究上述建模问题,从而提高实时计算过程中能量利用效率,本文用数据建模方法和模型预测策略相结合,建立集群全局范围的能耗控制模型.通过生产过程中的实际数据验证,其控制精度能够满足实际要求.

2 相关工作

2.1 大规模数据流计算

大规模数据流是一种存在偏序关系的数据流集合,可形式化表示为 $\Omega = \{S^1, S^2, \dots, S^n\}$, S^i 表示数据流, S^i 与 S^{i+1} 存在偏序关系,并且有两个重要的属性:(1)任务类型的复杂性^[6];(2)数据流任务模型的非平稳性.大规模数据流根据负载变化的剧烈程度可分为三种状态:正常态、高峰态和突发态.

2.2 GPU 集群能耗特征

若两个数据流集合 $\Gamma_i = \{S_i^1, S_i^2, \dots, S_i^m\}$ 和 $\Gamma_j = \{S_j^1, S_j^2, \dots, S_j^n\}$, $\Gamma_j \neq \Gamma_i$ 且 $\sum_{i=1}^m T_i^i = \sum_{j=1}^n T_j^j$ (T_i^i 表示 Γ_i 中的数据流 S_i^i 的计算时间).由于数据流任务模型的不确定性,导致各种数据流的计算能耗不同^[9].GPU 集群实时计算任务集为 $\Gamma = \{\Gamma_1, \Gamma_2, \dots, \Gamma_k, \dots\}$, 则计算能耗的变化轮廓可表示 $E = \{E_1, E_2, \dots, E_k, \dots\}$.

大规模数据流的两个属性影响 GPU 集群的计算能耗.(1)数据流任务模型的复杂性和时域性导致能耗随时间动态变化,即 E 具有较强时域性;(2)数据流任务模型的非平稳性造成计算能耗的不确定性.

3 基准能耗控制模型

GPU 集群中能量管理的思路如下: E_p 表示预先设置的能量配给值; E_o 为计算时实际能耗值, E_o 随着计算负载的波动而变化,具有一定的不确定性.若在计算过程中不断调整各时间段内的 E_p , 能量预测误差 $e_o = E_p - E_o$, 能量优化控制的目标就是保证计算性能的前提下尽量降低 e_o . 下面以两个典型的 CPU 集群能耗控制模型修改后作基准对比.

3.1 动态阈值控制模型

动态阈值调整法(Dynamic Threshold Scaling Model, DTSM)^[9]. 集群分为 m 个组进行控制管理,对于每个分组 g_i , 当各组实际能耗 E_o 低于阈值 θ 时,则提高该组内 GPU 利用率最大节点的能量配额;当各组实际能耗高于阈值 θ 时,则降低该组内 GPU 利用率最小节点的能量配额.

3.2 单输入输出控制模型

本节以文献^[10]改良的单输入输出控制(Improved Single Input Single Output, ISISO)作为基准模型.在计算中以固定控制律进行能耗控制,当各组实际能耗 E_o 高于

阈值 θ 时,管理者降低组内 GPU 利用率最大节点的能量配额和提高利用率最低节点的能量配额;当各组实际能耗 E_o 小于阈值 θ 时,管理者提高组内 GPU 利用率最大节点的能量配额和降低利用率最低节点的能量配额.

4 GPU 集群模型预测控制

上述两种方案从样本数据中训练学习,并预测未知数据的趋势,以此适应实时计算中能耗变化的时域性.然而实时计算中的任务模型具有非稳定性,其概率分布随时间变化导致功耗变化存在扰动.因此基准模型的自适应能力仍有局限性.本节提出一个既考虑历史能耗数据,又兼顾预测反馈信息的控制模型,核心策略是模型预测控制(Model Predictive Control)^[12],简记为 MPC.模型预测控制包括三部分:建立预测模型、滚动优化、反馈校正.

4.1 控制系统

图 1 为 MPC 能耗控制系统的关键模块:控制模块、测量模块、状态监测模块.控制系统的工作机制为:测量模块采集各节点实际能耗 $CP(K)$, 以此作为控制变量,传给控制器模块;状态监测模块中的性能监测器(Utilization Monitor, UM)记录节点的利用率并发送给控制模块;在控制模块中关于能耗和性能的历史数据 H 被传输到模型预测单元,利用上一个控制周期的反馈信息滚动优化预测模型的参数,优化后的控制模型再更新主控制器,最后发出控制指令 F 给执行单元,由执行单元调整各节点的能耗状态.

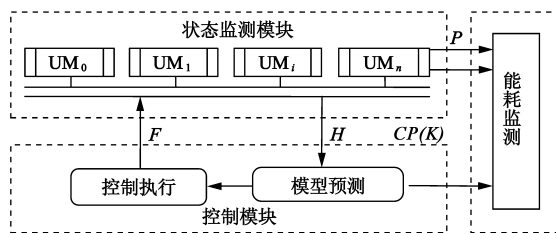


图1 GPU集群能耗控制系统

4.2 系统模型

控制模块的基础是系统模型,即操作变量与被控变量的关系模型,在此用系统辨识法建立系统模型.调整 GPU 能耗状态有两种方式:调整计算频率^[13]和改变活跃流多处理器(Stream Multiprocessor, SM)^[14], 都以牺牲计算性能为代价来降低 GPU 功耗,因此操作变量选择计算频率和活跃 SM.被控变量是集群的总能耗.因为操作与被控变量是关于时间的函数,所以系统模型是操作变量与被控变量的关系方程.假设计算过程为整个控制阶段,并以控制周期为单位来划分.表 1 为符号表.

首先考虑调整能耗状态对能耗的影响,在此用功率来量化计算能耗,并建立 GPU 功耗模型.功率与核心计算频率呈线性关系是经典 CPU 功耗模型^[5].从体系结构角度上 GPU 与 CPU 不同, GPU 简化计算单元的逻辑复杂度,增加芯片上计算单元的密度.因此 GPU 能耗模型与 CPU 稍有区别,计算功率主体与核心频率正相关,而活跃计算单元的数目对功率有些影响,随着活跃 SM 增加计算功率相应提高^[15].接下来建立计算频率、活跃 SM 与 GPU 计算功率的关系模型.

$$P_i(k)^f = a_i f_i(k) + b_i \quad (1)$$

式(1)表示节点 i 的功率是计算频率的近似线性函数, k 表示控制周期的第 k 阶段,参数 a_i 与节点类型有关, b_i 是调节参数, f 上标表示核心频率对功率的影响,以区别活跃 SM.其次建立活跃 SM 对功率调整部分的线性关系式^[6].

$$\Delta P_i(k)^{sm} = c_i SM_i(k) + d_i \quad (2)$$

式(2)中 c_i 和 d_i 是两个线性参数, $\Delta P_i(k)^{sm}$ 为改变活跃 SM 数量后计算功率的变化量.综合式(1)、(2)得出能耗状态调整与功率的关系式,

$$P_i(k) = a_i f_i(k) + b_i + c_i SM_i(k) + d_i \quad (3)$$

在式(3)基础上递推 $k+1$ 控制阶段的功率关系,

$$P_i(k+1) = P_i^f(k) + a_i \Delta f_i(k) + c_i \Delta SM_i(k) \quad (4)$$

再把式(4)用矩阵形式扩展到集群范围中,如下:

$$P(k+1) = P(k) + A \Delta f(k) + C \Delta SM(k) \quad (5)$$

其中 $P(K) = [p_1(k), p_2(k), \dots, p_n(k)]^T$, $A = \text{diag}[a_1, a_2, \dots, a_n]$, $C = \text{diag}[c_1, c_2, \dots, c_n]$.在功率初始值基础上用式(5)可推算整个控制过程中计算功率值.接下来考虑被控变量,设有 N 个节点,

$$CP(K) = \sum_{i=1}^N P_i(k) \quad (6)$$

将式(3)代入式(6)后得出能耗控制的系统模型,作为整个控制器的基础.

表 1 控制符号表

符号	描述
T	控制周期
$P_i(k)^f$	在 k 控制周期节点 i 的功率(频率部分)
$f_i(k)$	在 k 控制周期节点 i 的频率
$P_i(k)^{sm}$	在 k 控制周期节点 i 的功率(SM 部分)
$P_i(k)$	在 k 控制周期节点 i 的功率
$CP(k)$	在 k 控制周期集群的功耗
$\Delta f(k)$	在 $k+1$ 与 k 周期之间频率差值
$\Delta SM(k)$	在 $k+1$ 与 k 周期之间活跃 SM 的差值
P_s	能耗理想设定值

4.3 控制器

控制器包括系统模型,代价函数,参考轨迹和最小二乘法解法器.系统模型如上节所示.

$$V(k) = \sum_{i=1}^p \|cp(k+i|k) - \text{ref}(k+i|k)\|_{w(i)} + \sum_{i=0}^M \|\Delta f(k+1|k) + f(k+i|k) + \Delta sm(k+1|k) + sm(k+i|k) - F_{\max} - SM_{\max}\|_{R(i)}^2 \quad (7)$$

代价函数是量化控制器的预测值与理想参考值差距的函数^[5],以此来判断是否启动能耗状态调整.定义如式(7)所示, P 表示预测周期, M 表示控制周期, $W(i)$ 为误差权矩阵, $R(i)$ 为惩罚权矩阵.第一部分是预测值与理想参考值的差,即代价函数的主体部分,其中理想输出参考值由式(8)得到.

$$\text{ref}(k+i|k) = P_s - e^{-\frac{gT}{T_{\text{ref}}}} i (P_s - cp(k)) \quad (8)$$

理想输出参考函数产生一条功率变化轨迹,在整个控制过程中用其纠正实际输出值.若当前总功率高于参考值,则降低能耗状态,反之则提高能耗状态.式(8)为常见的指数函数形式,其中 T_{ref} 是系统响应时间常量,影响系统收敛速度.式(7)第 2 部分用于抑制控制增量,防止系统超过限定范围或发生剧烈振荡.另外,误差权矩阵和惩罚权矩阵用来调整控制输入与原始值的偏移量.最后控制器利用最小二乘法求解控制模型中的参数.

控制器的工作流程如下:在预测时域中控制器用系统模型来预测未来控制行为,在此指的是如何调整集群中各节点的核心频率和活跃 SM 数.然后选择代价函数最小,并且满足约束条件的预测控制行为作为控制输入,并得到一个预测控制行为,并与前一个控制周期的反馈信息 $CP(k)$ 结合,在反馈信息的基础上重新计算输入控制,并实施该控制周期的控制行为,反复迭代此过程直至控制结束.

5 实验结果

采用仿真和工程数据进行实验.仿真数据可灵活控制数据流的变化趋势,适合比较三种控制模型的精度;工程数据则验证在实际生产过程中的效果.仿真实验床是 16 个节点的 GPU 集群,为更好地实施基准控制模型, GPU 集群划分为 4 组,各组中 GPU 类型不同.

5.1 实验方法

实验思路是测量 GPU 集群的实际功耗,比较能耗控制中实时功率与设定值的偏离程度.计算过程中的实际能耗与理想设定值的偏离越小,说明控制精度越高.用自行设计的功耗卡来测量 GPU 能耗^[9],能耗状态通过调整核心频率和活跃 SM 数量两种方式来改变.

然而本文假设不考虑能耗状态切换造成的性能和能耗损失,采用下述方法来解决.假设一个计算任务 R 在 GPU 集群中的计算时间为 T ,根据控制周期将 T 划

分成一个连续的时间段序列,即 $T = \{t_1, t_2, t_3, \dots, t_k\}$. 通过调整计算频率和活跃 SM 来改变能耗状态,将计算频率分 4 个等级,活跃 SM 数分饱和、半饱和和未饱和 3 个等级,组合产生 12 种 GPU 的能耗状态,则能耗调整方案集合 $M = \{m_1, m_2, m_3, \dots, m_{12}\}$. 其中调整计算频率对功耗控制模型影响较大,而调整活跃 SM 的作用较小.再用每种调整方案计算 R ,每种方案记为 $P^{mi} = \{p_1^{mi}, p_2^{mi}, p_3^{mi}, \dots, p_k^{mi}\}$, P^{mi} 表示能耗调整方案 m_i 计算中的实际功率测试序列集合, P_i^{mi} 表示在执行 m_i 方案 t_i 时的功率值. 在计算 R 过程中控制模型实时调整能耗状态,即在同时段内应用不同调整方案.若计算任务 R 的控制模型的理论输出轨迹为 $R_m = \{m_3, m_5, m_{l-2}, \dots, m_8\}$, R_m 表示在 t_1 用 m_3 方案,在 t_2 时调整为 m_5 方案, t_3 时调整为 m_{l-2} ,直到 t_k 执行 m_8 结束. 因此计算 R 的实际功率为 $P_r = \{p_1^{m3}, p_2^{m5}, p_3^{m_{l-2}}, \dots, p_k^{m8}\}$. 该数据获取方法解决当前 GPU 硬件局限性造成的问题,具有计算精确度高的优点.

5.2 控制精度

先生成仿真数据,从 Nvidia 公司的 CUDA Demo 和文献[14]提供的基准程序中选择计算密度不同的程序集 $R = \{R_1, R_2, \dots, R_i\}$,再用高斯分布的随机策略来设定 R_i 的执行顺序,重复执行程序集 R 保证集群计算时间大于 3600s. 能耗控制系统的理想设置点 Set Point 为 4150W,数据图中用虚直线表示理想设置点.

图 2(a) 显示 DTSM 的能耗值与设定值的偏离最

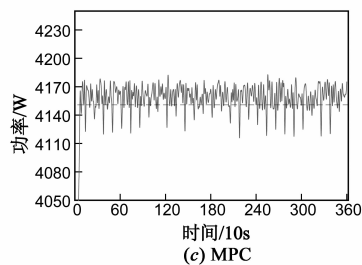
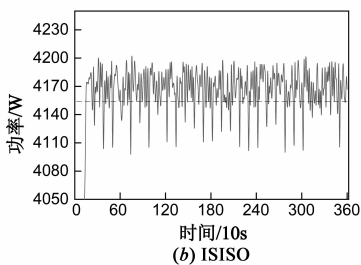
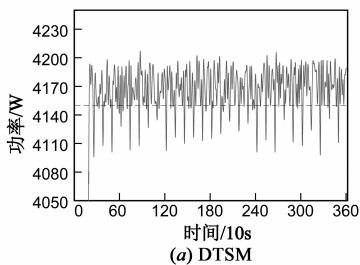


图2 三种能耗控制模型运行时能耗变化图

5.3 有效性

能耗控制模型的有效性可从两个方面评价:节能和控制稳定性.利用 4.2 节的结果来考察三种方法的节能指标.由于计算时间差别微小,因此用计算过程的平均功率来衡量节能效果.具体统计结果为:DTSM 是 4190.2W,ISISO 是 4189.7W, MPC 是 4160.3W.由此可见仿真实验中 MPC 的节能效果要优于前两个模型.

再用工程数据来验证控制稳定性,既控制模型能否较好的收敛在设定值范围内.工程数据分别为上海市电信 3G 视频质量检测系统^[16]和林肯实验室的入侵

大,波动范围大约为 $[-55W, +60W]$. 另外 DTSM 从启动到达理想设定值的时间最长,在实验中约为 280s. 由于 DTSM 能耗高于或低于设定值后需要较长时间向正确方向修正,存在控制滞后性.因为当能耗偏离设定值后才引发能耗状态的调整,导致滞后 2~3 个控制周期来调整能耗状态,所以实验中能耗值波动范围最大.此外,由于在集群分组范围内 DTSM 存在随机性,系统启动后收敛到稳定状态的时间最长.

图 2(b) 显示 ISISO 能耗值与设定值的偏离比 DTSM 小,波动范围大约在 $[-50W, +48W]$,从启动到达设定值的时间约为 210s.因为在 ISISO 中预先根据节点类型均匀分配能量,所以相比 DTSM 而言,减小了控制滞后性的影响范围,偏离的波动范围也相应减小.以上结果表明 ISISO 控制精度高于 DTSM. ISISO 以单个计算节点作为控制目标,将 DTSM 系统全局随机性改为局部随机性,提高了系统收敛速度,因此从系统启动到达设定值的时间明显小于 DTSM.

图 2(c) 显示 MPC 能耗值与设定值的偏离最小,波动范围大约在 $[-30W, +33W]$,从启动系统到达设定值的时间约为 50s.由于用历史数据预测未来控制行为和滚动优化机制,保证控制系统的滞后性最小,控制精度明显好于前两种模型.另一个方面,控制模型启动的时间最短,则说明 MPC 的收敛速度最快.由于在设置理想设定值时未考虑计算中额外的能量损失,因此大多数实际测量值大于设定值.

检测数据.由于 3G 视频流和网络数据流具有明显正常、高峰、突发三种状态,仿真数据难以模拟该特性,因此工程数据可检验控制系统的应用价值.3G 视频流的原始形式是 RTP 数据包, GPU 负责 H.264 数据还原、视频特征提取等数据密集任务.入侵检测数据基本形式是 Tcpdump 格式的网络数据包, GPU 处理字符串过滤分析.在此不考虑负载不均衡的问题,由人工方式实现数据流分配.

图 3 显示视频质量检测计算中一个小时的能耗控制情况,能耗设定值为 4000W,从系统启动到达稳定状

态大约 300s,与设定值的偏离大约为 $[-32W, +35W]$. 由于视频质量检测任务的计算密度比较稳定,而且数据流装载时间基本呈正态分布,因此能耗被稳定地控制在设定值范围内,仅出现7次稍大偏离.此外,工程测试中的数据流装载和分配的时间较长,导致系统到达稳定态的时间比仿真数据的实验长.

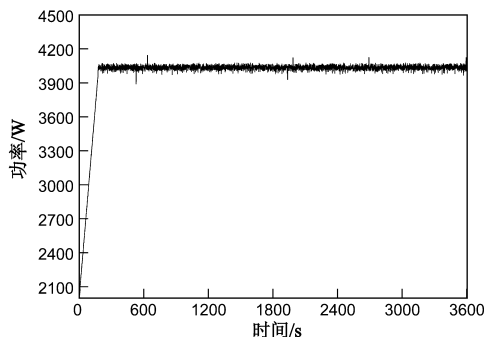


图3 MPC控制视频质量检测计算过程

图4显示入侵检测计算中一个小时的能耗控制情况,能耗设定值为3900W,从系统启动到达稳定状态大约500s,与设定值的偏离基本保持在 $[-46W, +57W]$. 因为入侵检测计算中网络数据包的大小变化复杂,数据流装载的不确定性较大,所以导致出现6次较大波动的偏离,而且存在时间稍长.这说明在处理长度非常不规则数据流计算时,MPC控制模型仍存在扰动,但是基本保证了控制稳定性.

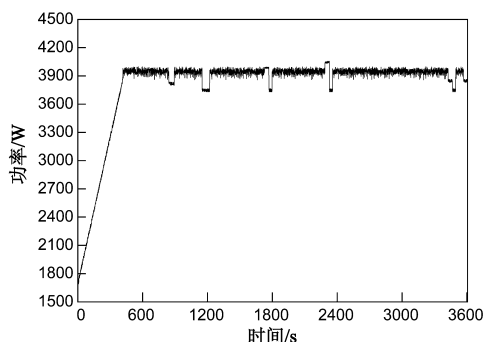


图4 MPC控制入侵检测计算过程

5.4 不确定性

本节验证 MPC 对能耗变化不确定性的适应能力. 由于计算能耗的波动与计算负载的变化存在较强相关性,在此用计算负载的变化来模拟能耗变化的不确定性,通过逐渐增加程序集 R_i 中程序的数量来仿真计算负载的变化. 例如,开始 $t = 5$,此时只用5个程序随机组合产生仿真数据流.随着 t 值的增加,仿真数据流任务模型的复杂性逐渐增加,导致能耗变化的不确定性也逐渐增加.如图5(a)显示三种控制模型运行30分钟的能耗对比,随着能耗变化不确定性的提高,系统能耗

逐渐增加,其中 MPC 增幅最小,DTSM 增幅最大.这项结果表明,MPC 适合处理实时变化的数据流,在集群范围内能够根据负载变化来优化能耗.由于 ISISO 运用一定反馈控制机制,随着计算负载变化,能耗波动不如 DTSM 剧烈.

在工程数据实验中,选择视频质量检测系统中三个时段的数据流:8:00-9:00、15:00-16:00 和 17:00-18:00,分别代表高峰期、正常期和突发期数据流.其中选择有体育竞赛的 17:00-18:00 时段作为突发期,由于在该时段年轻用户观看视频引起流量产生异常变化.如图5(b)所示,以 DTSM 与 ISISO 模型控制时,高峰期和突发期视频流的能耗明显高于正常期能耗.而 MPC 模型流量突发期与正常期能耗相似,突发期并不是视频流量剧增,而是流量变化剧烈,MPC 模型适合处理计算负载动态变化的任务,因此能耗并无明显改变,只是在高峰期计算负载普遍增加后能耗才表现出比正常期增大.

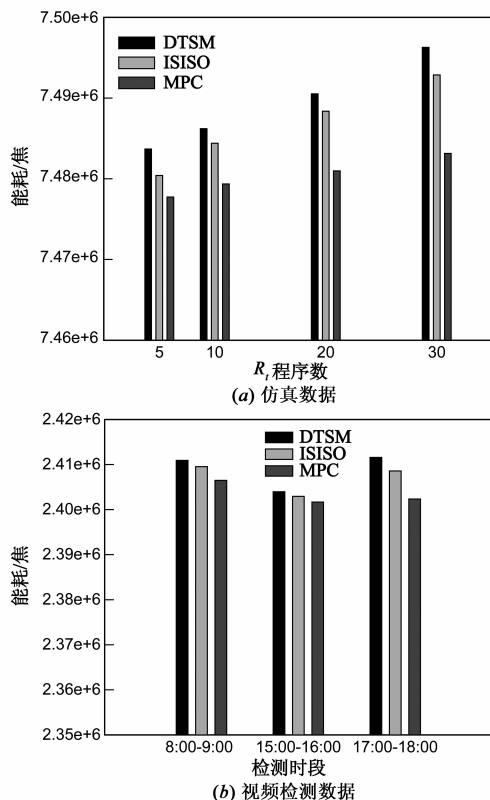


图5 计算负载具有不同不确定性时的能耗对比

5.5 参数优化

本节用实验方法寻找较优的控制周期.为了降低实验复杂性只选用4个节点执行入侵检测计算,能耗理想设定值为990W,观察能耗实测值与设定值的偏离情况,根据偏离程度来决定控制周期.选择0.1s、1s、10s三个数量级的控制周期进行实验,都选择600个控制周

期,系统运行时间为 60s,600s,6000s.

图 6(a)显示控制周期为 0.1s 时, GPU 集群系统运行 60s 的状态图. 初期出现一些较大的正偏离, 然后逐渐稳定, 未见明显抖动现象. 图 6(b)显示控制周期为 1s 时, GPU 集群系统运行 600s 的状态图. 能耗情况一直比较平稳, 但是与图 6(a)相似, 能耗实测值均明显高于设

定值. 图 6(c)显示控制周期为 10s 时, GPU 集群系统运行 6000s 的状态图. 能耗不仅一直趋于平稳, 而且偏离波动最小. 与图 6(a)、(b)不同, 能耗实测值在设定值上下波动. 此外, 计算节点的能耗状态调整的间隔越长, 调整引起的计算性能损失越小. 因此选择 10s 作为控制周期能保证系统的控制精度和稳定性.

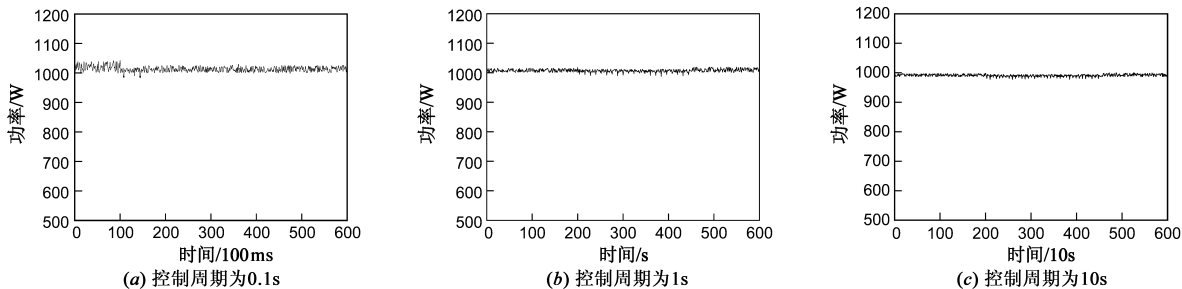


图6 不同控制周期的能耗对比

6 结论

本文重点关注计算负载的不确定性和系统稳定性两种因素, 从集群全局角度建立一个可实现能耗优化和消减的控制模型, 利用历史数据判断未来控制行为, 同时用前一个控制周期的总功耗作为反馈信号来调整预测值, 通过滚动优化的方法动态调整控制行为, 克服了静态能量分配方案的盲目性. 另一方面, 该能耗控制模型属于多输入多输出模型, 解决单输入单输出只关注局部优化的局限性. 在与 DTSM 和 ISISO 对比实验中显示, 模型预测的控制精度高, 节能效果好, 而且稳定性好. 研究中存在一些细节问题尚未解决, 如 GPU 集群理想设定值的确定还需要进一步理论分析, 未来尚且需要探索两种能耗状态调整方式对功耗控制模型的定量影响.

参考文献

- [1] 杨正龙, 金林, 李蔚清. 基于 GPU 的图形电磁计算加速算法[J]. 电子学报, 2007, 35(6): 1056 - 1060.
Yang Zheng-long, Jin Lin, Li Wei-qing. Accelerated GRE-CO based on GPU[J]. Acta Electronica Sinica, 2007, 35(6): 1056 - 1060. (in Chinese)
- [2] 刘勇鹏, 王锋, 卢凯, 等. 面向异构并行计算系统的流水线式压缩检查点[J]. 电子学报, 2012, 40(2): 223 - 229.
Liu Yong-peng, Wang Feng, Lu Kai, et al. Pipelined compressed checkpointing for heterogeneous systems[J]. Acta Electronica Sinica, 2012, 40(2): 223 - 229. (in Chinese)
- [3] Horvath T, Abdelzaker T, Shadron K, et al. Dynamic voltage scaling in multitier web servers with end-to-end delay control[J]. IEEE Transaction on Computers, 2007, 56(4): 444 - 458.
- [4] Bertini L, Julius C B, Mosse D. Power optimization for dynamic

- configuration in heterogeneous web server clusters[J]. Journal of Systems and Software, 2010, 83(4): 585 - 598.
- [5] Wang Xiao-rui. Coordinating power control and performance management for virtualized server clusters[J]. IEEE Transaction on Parallel and Distributed System, 2011, 22(2): 245 - 259.
- [6] 林一松, 杨学军, 唐滔, 等. 一种基于并行度分析模型的 GPU 功耗优化技术[J]. 计算机学报, 2011, 34(4): 705 - 716.
Lin Yi-song, Yang Xue-jun, Tang Tao, et al. A GPU low-power optimization based on parallelism analysis model[J]. Chinese Journal of Computers, 2011, 34(4): 705 - 716. (in Chinese)
- [7] 林一松, 杨学军, 唐滔, 等. 一种基于关键路径分析的 CPU-GPU 异构系统综合能耗优化方法[J]. 计算机学报, 2012, 35(1): 123 - 133.
Lin Yi-song, Yang Xue-jun, Tang Tao, et al. An integrated energy optimization approach for CPU-GPU heterogeneous system based on critical path analysis[J]. Chinese Journal of Computers, 2012, 35(1): 123 - 133. (in Chinese)
- [8] 赵霞, 郭耀, 雷志勇, 等. 基于模拟器的嵌入式操作系统能耗估算与分析[J]. 电子学报, 2008, 36(2): 209 - 215.
Zhao Xia, Guo Yao, Lei Zhi-yong, et al. Estimation and analysis of embedded operating system energy consumption[J]. Acta Electronica Sinica, 2008, 36(2): 209 - 215. (in Chinese)
- [9] 王海峰, 陈庆奎. 静态程序切片的 GPU 通用计算功耗预测模型[J]. 软件学报, 2013, 24(8): 1746 - 1760.
Wang Hai-feng, Chen Qing-kui. Power consumption prediction model of general-purpose computing GPU with static program slicing[J]. Journal of Software, 2013, 24(8): 1746 - 1760. (in Chinese)
- [10] Wang Hai-feng, Chen Qing-kui. Power estimating model and analysis of general programming on GPU[J]. Journal of Soft-

ware, 2012, 7(5): 1164 – 1170.

- [11] Tang Qing-hui, Gupta S K, Georgios V. Energy-efficient thermal-aware task scheduling for homogeneous high-performance computing data centers: A cyber-physical approach [J]. IEEE Transaction on Parallel and Distributed System, 2008, 19(11): 1458 – 1472.
- [12] 徐祖华, 赵均, 钱积新. 基于多自由度性能指标的模型预测控制算法 [J]. 电子学报, 2008, 36(5): 906 – 909.
Xu Zu-hua, Zhao Jun, Qian Ji-xin. An improved model predictive control algorithm based on multi-degree-of freedom performance index [J]. Acta Electronica Sinica, 2008, 36(5): 906 – 909. (in Chinese)
- [13] Yang Xue-jun, Yan Xiao-bo, Xing Zuo-cheng, et al. Fei teng 64 stream processing system: architecture, compiler, and programming [J]. IEEE Transactions on Parallel and Distributed Systems, 2009, 20(8): 1142 – 1156.
- [14] Hong S, Kim H. An analytical model for a gpu architecture with memory-level and thread-level parallelism awareness [A]. Proceedings of ISCA '09 [C]. New York: ACM, 2009. 152 – 163.
- [15] Hong S, Kim H. An integrated GPU power and performance model [A]. Proceedings of ISCA '10 [C]. New York: ACM, 2010. 280 – 289.
- [16] 王海峰, 陈庆奎. 图形处理器通用计算关键技术研究综述 [J]. 计算机学报, 2013, 36(4): 757 – 772.

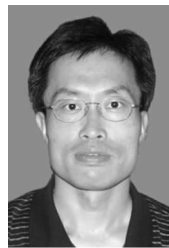
Wang Hai-feng, Chen Qing-kui. General purpose computing of graphics processing unit: a survey [J]. Chinese Journal of Computers, 2013, 36(4): 757 – 772. (in Chinese)

作者简介



王海峰(通讯作者) 男, 1976年3月出生, 山东临沂人. 临沂大学副教授, ACM 会员. 1997年、2005年和2012年分别在山东大学、石油大学和上海理工大学获计算机应用专业的工学学士、工学硕士和理学博士学位. 主要研究方向为高性能计算、智能控制.

E-mail: gadfly7@126.com



曹云鹏 男, 1967年10月出生, 山东沂南人. 临沂大学副教授. 1989年和2005年分别在南开大学和山东科技大学获得计算机应用专业的学士和硕士学位, 主要研究方向为并行计算、智能控制和仿真.

E-mail: lyucyp@163.com