

一种基于主动认知决策的高效能模型

杨 劲^{1,2} 庞建民^{1,2} 王俊超¹ 于锦涛¹ 刘 睿³

(解放军信息工程大学 郑州 450001)¹ (数学工程与先进计算国家重点实验室 郑州 450001)²
(解放军 61345 部队 西安 710100)³

摘要 随着高性能计算机的发展,如何降低功耗、提高计算机的效能,越来越受到人们的关注。针对高性能计算机的效能问题,采用可重构思想,提出一种基于主动认知决策的高效能模型。该模型不断感知应用任务的实时状态,并对应用状态和当前应用结构做出评估决策,通过主动重构应用结构来达到降低系统能耗和提高效能的目的。为了验证模型的有效性,搭建了原型实验系统,实现了视频拷贝检测和口令恢复两个应用,并采用真实的互联网流量统计曲线模拟负荷。结果表明,基于该模型构建的应用能够提高系统的效能,在上述环境下,与传统方法相比,效能提高了 58%。

关键词 高效能,低能耗,高性能,可重构计算,主动认知,结构决策

中图分类号 TP311 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.11.014

High-productivity Model Based on Proactive Cognition and Decision

YANG Jin^{1,2} PANG Jian-min^{1,2} WANG Jun-chao¹ YU Jin-tao¹ LIU Rui³

(PLA Information Engineering University, Zhengzhou 450001, China)¹

(State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450001, China)²

(Unit 61345, Xi'an 710100, China)³

Abstract With the development of HPCs, increasingly importance has been attached to reducing power consumption and raising productivity. We proposed a high productivity computing model to deal with the HPCs' productivity problem, which adopts the concept of reconfigurable computing and is based on proactive cognition and decision system. This model apperceives the real-time states of application tasks, evaluates the matching degree of application state and current application structure, and then reconfigures the application structure to lower energy consumption and increase productivity. In order to verify the model's effectiveness, we constructed a prototype experimental platform, implemented a video-copy-detection program and a password recovery program, and then used real Internet traffic statistical curve to simulate the programs' loads. Experimental results demonstrate that under this environment the application system based on the model has raised its productivity by 58% compared with traditional method.

Keywords High productivity, Energy-efficient, High performance, Reconfigurable computing, Proactive cognition, Structure decision

1 引言

近年来,基于计算机软硬件技术的发展和高性能计算的应用需求,HPC的浮点计算峰值能力获得了稳定增长,基本保持每 10 年提高 1000 倍的速率。然而,在 HPC 峰值性能不断刷新记录带给人们惊喜的同时,我们也不得不黯然接受两个令人沮丧的事实:(1)计算机功耗越来越大;(2)计算机的实际应用效率并没有获得期望的提升。

一方面,计算机的功耗居高不下,运行和维护费用较高。2013 年 Top500 排行榜排名第一的天河 2 号计算机的功率为

17.8MW,以目前的电价计算,它运行一年的电费大约为 1.5 亿元人民币。另一方面,计算机的应用效率没有本质性的提升。多数高性能计算机的 Linpack 运算速度达到其峰值速度的 60%甚至 80%以上,而实际应用任务的运算速度仅为峰值速度的 10%左右,某些应用任务甚至更低^[1]。

造成上述问题的原因有很多,我们主要关注两个问题:(1)系统中的处理设备在无负载时处于空转状态或者同时存在多个较低负载的处理设备。通常,高性能计算机为满足峰值应用配置了大量冗余硬件,而在实际应用中,应用任务并不能完全利用所有的冗余硬件,造成大量的冗余硬件处于空转

到稿日期:2014-11-20 返修日期:2015-01-20 本文受上海市科委科技攻关项目(13DZ1108800),国家高技术研究发展计划(2009AA012201),国家自然科学基金(61472447)资助。

杨 劲(1983-),男,博士生,主要研究方向为高性能计算、可重构计算,E-mail:ysire@163.com;庞建民(1964-),男,博士,教授,主要研究方向为高性能计算、信息安全;王俊超(1988-),男,博士生,主要研究方向为高性能计算、动态编译;于锦涛(1987-),男,博士生,主要研究方向为可重构计算、高层次综合。

或低效使用状态,从而造成能源浪费。(2)应用任务的资源需求与资源调度之间的不匹配,具体包括用户需求的资源难以得到保证,以及由于调度不当把低效的资源分配给了应用任务等。高性能计算机往往包含多种类型的计算和处理资源,如通用 CPU、FPGA、GPU 等。由于制造工艺和面向的应用领域不同,各类器件擅长处理的应用类型也各不相同。对于同一个应用任务,选用不同类型的计算资源,处理性能、资源和能源消耗等各不相同,计算效能也往往不同。

本文面向高性能计算机的效能问题,采用认知决策的思想,提出一种基于主动认知决策的高效能模型。该模型动态感知应用任务的运行态势,从提高计算机系统利用率和降低系统功耗入手,根据应用任务需求和系统的运行结构做出预先决策,主动重构高性能计算机系统的运行结构,使系统提供与应用任务需求相匹配的工作状态,达到在满足应用性能的前提下系统能耗最小的目的。实验表明,基于该模型构建的系统既能够为应用任务提供高性能服务,也可以有效降低系统整体能耗,提高计算机的效能。

2 系统框架

根据文献[5],在混合可重构集群系统中,组织各类处理单元的方式大致可以分为两类:一致节点非一致系统(Uniform Node Nonuniform System, UNNS)或者非一致节点一致系统(Nonuniform Node Uniform System, NNUS)。

我们发现,从结构上看,UNNS 和 NNUS 的区别仅在于处理单元的通信方式和节点内处理单元的相互关系。UNNS 中每个处理单元都通过通信总线相互通信,节点内处理单元既可以紧耦合,也可以松耦合。而 NNUS 中,节点间处理单元通过通信总线传输数据,而节点内处理单元通过系统总线传输数据;节点内处理单元通常是紧耦合关系。为了提高通信性能并降低处理单元之间的耦合度,在 UNNS 和 NNUS 的基础上提出一种混合可重构集群系统的组织方式:非一致节点非一致系统(Nonuniform Node Nonuniform System, 3NS)。图 1 给出了 UNNS、NNUS 和 3NS 结构的模型示意图。

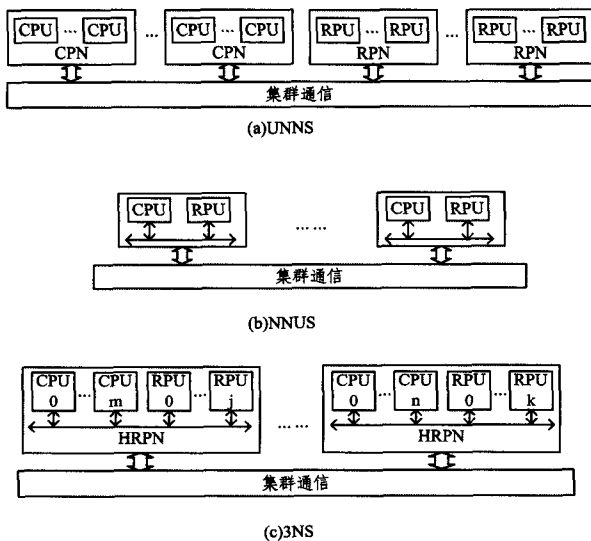


图 1 异构可重构集群的组织方式

如图 1(c)所示,从节点的层次看,3NS 每个节点同时包

含 CPU 和可重构单元,是非一致的;从系统的层次看,每个节点也都不完全相同。3NS 不再严格约束节点的具体组成,节点内的处理单元不存在主从关系。在节点内部,CPU 和非 CPU 处理单元的比例可以自由调整。理论上,当节点内 CPU 个数或 RPU 个数的其中一个为 0 时,该节点将退化为一致的;当 CPU 个数和 RPU 个数都不为 0 时,该节点为非一致的。在系统层面,当所有节点组成恰好相同时,系统将退化为一致的;反之,系统是非一致的。因此,从这个角度上说,UNNS 和 NNUS 只是 3NS 的特殊情况。需要指出的是,尽管图 1 和本文实验平台仅使用了可重构 FPGA 作为异构设备,但实际上 3NS 也可以包含其它类型的异构设备。

图 2 给出了实验平台的总体架构。原型实验平台中一共包含了 15 个节点,它们又被分为 2 类:10 个通用处理节点(CPN)和 5 个混合可重构处理节点(HRPN)。使用 IBM 的 X3650 M3 商用服务器作为 CPN,每台机器包含一个 Xeon X5650 2.66GHz 的 6 核处理器,2 * 4GB 的内存,并配备 2 个 Intel 的 4 端口以太网千兆网卡 I350-T4 和一个 Intel 万兆网卡 82599EB。在 HRPN 中,每个节点包含 1 个 Atom 处理器、4 个 Xilinx Virtex-6 XC6VLX550T FPGA 芯片。Atom 处理器配备 2GB 内存。每个 FPGA 配备了独立的 3 * 8GB 的内存、1 个 Intel 万兆网卡 82599EB。节点内 FPGA 通过点对点的 GTX * 4 进行数据传输,节点间处理单元采用 InfiniBand 网络进行数据通信,对千兆以太网进行管理控制。Atom 处理器与 FPGA 通过 Jtag 菊花链的方式连接。

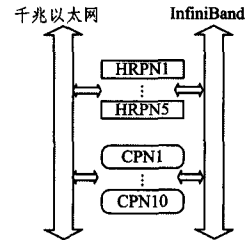


图 2 原型平台架构

3 高效能模型

本节描述一种基于主动认知决策的高效能模型。对应用任务 T ,假设它由子任务 T_1, T_2, \dots, T_n 组成,每个子任务分别由处理单元组 PE_{T_i} (可能包含多个处理单元)完成。该模型动态感知计算机系统对应用任务的处理情况,通过为每个子任务选择性能和能耗最匹配的处理单元,使得应用任务的运行结构总能与任务的处理情况匹配良好,从而达到系统的高性能和低能耗。

3.1 相关定义

定义 1(构件) 用于处理子任务、可被系统重构的基础处理单元称为构件,一般用 c 表示。

构件是应用任务和处理单元的结合体。单独的应用任务和单元都不是构件,只有当应用(子)任务被调度加载到某个处理单元或其中一部分上,并且该处理单元支持重构功能时,这个(部分)处理单元才被称为构件。子任务与构件是一对一或一对多的关系,每个子任务可以有一个或多个构件处理,但一个构件不能处理多个子任务,一个子任务的所有构件称为同功能构件。构件与处理单元之间并没有固定的对应

关系,虽然如此,仍把相同子任务在相同类型处理单元上的构件看成是同类型构件。同功能构件并不一定是同类型构件,而同类型构件一定是同功能构件。本文假定同类型构件的性能、功耗属性是相同的,并且是可测量或可估值的。

定义 2(结构) 指的是应用任务的运行时结构,即在应用任务运行的某个时刻,正在参与处理该任务的构件和它们的拓扑关系,通常记作 s 。

组成 s 的所有构件称为 s 的构件组,记作 G_s 。在 G_s 中,同类型构件可能有多个,它们在 s 中的拓扑层次可能相同,也有可能不同。去除 G_s 中所有重复的同类型构件后,剩余的构件集合称为 s 的构件基,记作 C_s 。类似地,去除 G_s 中重复的同功能构件后,剩余的构件集合称为 s 的基结构,记作 s_k 。

对应用任务 T ,所有结构组成的集合称为结构集,记作 S_T 。根据基结构定义,对任意 $s \in S_T, s_k$ 也是 S_T 其它元素的基结构。可以证明, S_T 是有穷的。在实际的计算系统中,可重构资源总是有限的,其总数记为 R 。设 $G_{S_i} = \{c_1, c_2, \dots, c_m\}$, c_i 消耗的资源为 r_i ,则系统能重构 c_i 的最大数目不超过 R/r_i ,记这个数目为 w_i 。所以, S_T 的元素个数 $|S_T| = \prod_1^m w_i$ 。

在结构集中,每个结构都可以用于处理应用任务 T ,它们的性能和能耗参数各不相同。结构可以相互转换,以匹配应用任务的负载、性能和能耗需求。认知决策的主要作用就是根据应用任务的实时负载变化,在众多的结构中选择一個能效比最好的方案。

定义 3(应用状态) 在应用任务处理的过程中,用来描述应用任务运行态势的一组值,通常用 q 表示。

应用状态反映应用任务运行的实时处理情况,根据认知的属性不同,应用状态的内容也不相同。在一个追求效能的 Web 服务器中,应用状态的一个可能描述为 $q = (conn, tp, se)$,其中 $conn$ 表示连接数, tp 表示吞吐量, se 表示会话数。

应用状态是感知数据的映射。感知器按照一定的时间周期从构件上获取感知数据,通常原始的感知数据不能直接用于认知决策(可能带有噪声等),需要把它们转换为应用状态。转换的方法有很多,既可以通过固定的函数转换,也可以通过复杂的机器学习方法建模后转换,需要根据实际情况采取合适的方法。在应用任务运行过程中,按照感知时间点,感知数据组成的序列称为感知数据表,记作 Σ ;应用状态组成的序列称为应用状态表,记作 Q 。

3.2 模型的形式化描述

定义 4 模型 M 是一个七元组:

$$M = (S, A, \Sigma, Q, \varphi, \delta, \lambda)$$

其中,(1) S 是有穷的结构集;(2) A 是有穷的动作集;(3) Σ 是感知数据表;(4) Q 是应用状态表;(5) $\varphi: \Sigma \rightarrow Q$ 是认知函数;(6) $\lambda: S \times Q \rightarrow R_0^+$ 是评估函数;(7) $\delta: S \times A \rightarrow S$ 是动作函数。

其中, R_0^+ 表示非负实数集。 S, Σ, Q 的意义已经做了说明,不再赘述。

动作表示从一个结构变换到另一个结构所需要的操作,本文主要讨论增加、删除构件,分别用 $+n, -n$ 表示,其中 n 为数目。假设结构 s_1 由 2 个 c_1 和 4 个 c_2 组成,表示为 $s_1 = \{2c_1, 4c_2\}$ 。结构 $s_2 = \{1c_1, 2c_2\}$,则从结构 s_1 变换为结构 s_2 的动作为 $a_{1 \rightarrow 2} = \{-c_1, -2c_2\}$ 。动作集 A 是所有动作的集合。可见,动作集的元素数目也是有穷的。

认知函数 φ 是一个从感知数据集到认知状态集的映射,负责把从感知器采集到的原始数据转换为能够直接参与决策评估的应用状态。评估函数 λ 通过评估系统结构和应用状态的契合度完成决策功能,其值域为非负实数集。函数的值越小,表明系统结构与认知状态的契合度越高;反之,表明系统结构与认知状态的契合度较低。

该模型的运行机制可以简述为:认知—决策—变换(实施动作)—认知—...,即一个从设备状态认知、决策选择最佳结构到设备应用动作变换的闭路循环过程。具体过程可以总结为:系统源源不断地通过感知器采集系统的运行数据,认知函数把感知数据转换为认知状态,评估函数对认知状态与系统结构进行评估和决策,得到新的结构,动作函数在现有结构上施加动作,把现有结构变换为新结构。如果把认知状态看作是应用程序运行状态的反映,评估函数看作是应用程序对系统资源的需求评估,则该模型实际上是一个根据应用程序实时需求不断变换自身结构的自适应模型。

4 实验评估

为了验证本文所提出的高效能模型的有效性,在原理验证系统上实现了视频拷贝检测和 word 2010 口令恢复两个应用,并对是否使用主动认知决策时的效能值进行比较。

4.1 实验设置

实验平台架构如图 2 所示,各节点组成如第 1 节所述。应用任务的主要构件如表 1 所列。应用任务的负载曲线模拟互联网数据流量在一天 24 小时中的变化情况^[15],如图 3 所示。从高效能角度看,希望投入服务应用的处理单元的数量能够随到达的负荷变化而增减,使空转的计算资源尽量少。

表 1 构件属性表

(a)视频拷贝检测

名称	功能	性能 (次/秒)	效能 (次/(瓦·秒))
similarity@CPU	CPU 上的相似度计算	15.2	0.103
similarity@FPGA	FPGA 上的相似度计算	1864.7	33.6

(b)word 2010 口令恢复

名称	功能	性能 (次/秒)	效能 (次/(瓦·秒))
Hash@CPU	CPU 上的 Hash 迭代	48.7	0.36
Hash@FPGA	FPGA 上的 Hash 迭代	2603.1	54.2
Firstest@CPU	CPU 上的一次验证	511709	43.8
Firstest@FPGA	FPGA 上的一次验证	33121609	3065
pwdExpend@CPU	CPU 上的口令扩展	341962	25.3
pwdExpend@FPGA	FPGA 上的口令扩展	26263815	2679

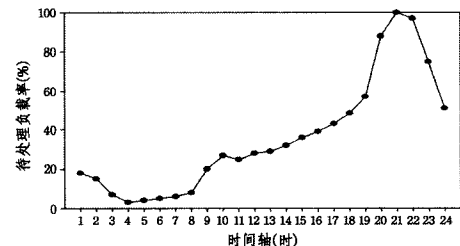


图 3 24 小时负荷曲线模型

视频拷贝检测应用采用 128 位精度表示特征,然后通过 64 帧聚合结果判断视频是否匹配。视频特征库由 1027 个高质量视频和 3623 个低质量视频共 4650 个视频 1510 万帧的

特征组成。待测视频是从 1027 个高质量视频中随机选取的 20 个视频,对每个视频分别进行台标、字幕、分辨率、亮度、对比度和宽高比等 6 种变化,得到 120 个待测视频。口令恢复应用的口令来源主要有两个:字典和穷举。字典库共包含 9600 个大小为 100M 的字典文件,存储于大容量盘阵上,通过万兆以太网与处理单元互联。穷举模式口令由处理单元内部产生。待破解文件共 1000 个,随着负载曲线变化循环使用。

效能评估标准基于绿色网络组织提出的 DCPE 定义。结合本实验实际情况,定义效能值为已经处理的任务数与消耗能源的比值,或者是性能与功耗的比值,可以证明,二者是等价的。

4.2 单节点最大性能和效能

测试的方法是向处理单元一次性发送尽可能多的处理任务,然后根据运行时间和耗电量得出单节点性能和效能。测试数据和结果如表 2 所列。可以看出,虽然 HRPN 节点的功耗高于 CPN 节点,但 HRPN 节点的性能和效能都远远高于 CPN 节点,这主要有 3 方面的原因:(1)一个 HRPN 有 4 个 FPGA 处理单元,而一个 CPN 只有 1 个 CPU 处理单元;(2)CPN 除了应用任务负载外,还要运行操作系统等额外负载;(3)FPGA 更适合用于并行处理。尽管 CPN 节点启动了 12 个线程,但其仍然与 HRPN 的性能和效能相差较多。

表 2 单节点最大性能和效能

节点类型	总负载量 (帧)	运行时间 (秒)	检测速度 (帧/秒)	总耗电量 (度)	平均功率 (瓦)	效能 (帧/(秒·瓦))
CPN	54960	3958.57	13.88	0.18	163.70	0.0848
HRPN	22689000	3659.93	6199.30	0.23	226.23	27.4022

(b) word 2010 口令恢复

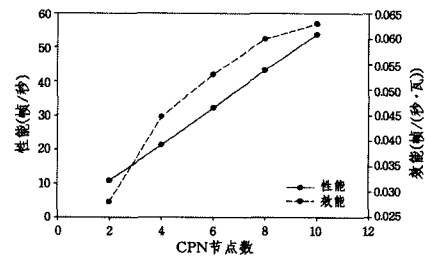
节点类型\破解方式	总口令数 (个)	运行时间 (秒)	性能 (个/秒)	总耗电量 (度)	平均功率 (瓦)	效能 (个/(秒·瓦))
CPN\字典	108011	3600	30	0.15	150	0.2
CPN\穷举	111603	3600	31	0.15	150	0.207
HRPN\字典	8746419	3600	6132	0.23	230	26.66
HRPN\穷举	28836137	3600	8010	0.19	190	42.15

4.3 集群平均性能和效能

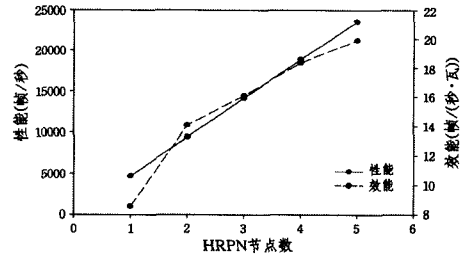
为了测试在不同系统规模下,高效能模型对两种应用任务性能和效能的影响,设计了视频拷贝检测、口令恢复字典模式和口令恢复穷举模式分别在 CPN 节点规模为 2、4、6、8、10 和 HRPN 节点规模为 1、2、3、4、5 时的 30 种实验场景。

测试结果分别如图 4、图 5 所示。可以看出,CPN 集群的性能并不是单个节点最大性能的倍数,与最大性能的倍数相差较多,这主要是由 CPN 节点重构时间较长和处理速度较慢造成的。HRPN 集群的少量性能损失主要是因为负荷发送模型导致部分节点在某段时间处于“饥渴”状态,没有发挥出最大性能。

对比表 2、图 4 和图 5 还可得出:(1)当处理单元重构时间较短、处理速度较快时,认知决策和系统重构取得的效果较好;(2)认知决策和系统结构重构并没有损失系统的额外性能;(3)当集群规模达到一定程度时,系统性能可以达到或接近系统的最大性能;(4)认知决策和系统结构重构对效能提升明显,尤其是对处理节点数比较多的大规模系统。

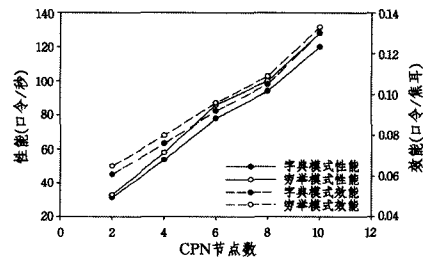


(a) CPN 节点性能和效能

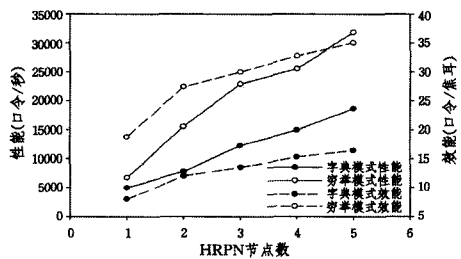


(b) HRPN 节点性能和效能

图 4 视频拷贝检测系统的平均性能和效能



(a) CPN 节点性能和效能



(b) HRPN 节点性能和效能

图 5 口令恢复应用系统的平均性能和效能

4.4 动静态结构效能对比

主动认知决策和系统结构重构是高效能模型的核心,对提高系统效能起着关键作用。它们主要通过两种方式产生作用:(1)在负载升高时,通过预先重构相关构件扩充系统处理能力,提高系统处理性能;反之,删除空转或空闲构件,降低能耗。即通过动态改变应用的运行时结构提高效能。(2)在异构系统中,通过选择和重构高效能的构件来代替低效能的构件,不断改善应用的运行结构,以提高效能。对于后者,通过图 4、图 5 可以看出,系统需要重构时,总是选择能效比最合适的构件,以保持系统的良好状态。本节主要阐述前者对效能的影响。

为了测试不同规模集群上的认知决策效果,实验对 CPN 节点分别为 2、4、6、8、10 个共 5 种场景进行了测试,测试以视频拷贝检测应用为例。

实验结果如图 6 所示。在动态服务器集群中,由于重构需要时间,容易导致性能损失。从图 6(a)可以看出,动态服务器集群的性能并没有明显损失,与静态服务器集群的性能

基本持平,说明认知决策和主动重构能够保证应用对性能的需求。另一个结论是,当服务器集群达到一定规模时,与静态服务器集群相比,动态服务器集群的效能提升比较明显。当节点规模较小时,服务器集群的能效值与单节点最大能效值相差较大;随着节点规模的增加,服务器集群的能效值将趋于稳定,如图6所示。动态服务器集群因为有认知决策系统支持,其能效值要高于静态服务器集群。在本实验的负荷模型下,当节点数为10时,动态服务器的效能提升了58%左右。

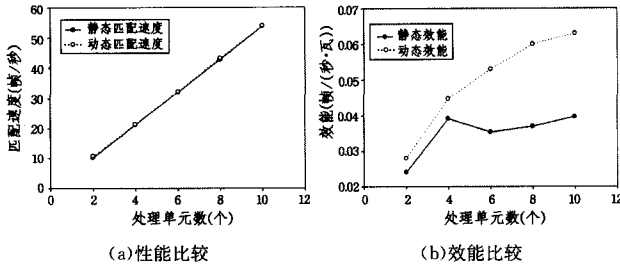


图6 系统动、静态性能和能效对比

5 相关工作

在已有的文献中,提高效能的途径有很多,大致分为体系结构、动态功耗管理、软硬件协同、虚拟化等几类。在体系结构方面,高效能系统更倾向于选择 GPU、FPGA 或其它能高度并行化的设备,Green500 2013 年排行榜上前 40 位的绝大多数都使用了 BlueGene/Q 架构。文献[2]对 GPU 和 FPGA 设备进行了比较,KH Tsoi 使用二者构建了一个异构系统 Alex^[7],并用 N-body 仿真实验对它们的性能进行了测试。文献[3,7,13,14]对动态功耗管理进行了研究,文献[4]提出使用在线学习的方法优化 DPM。文献[6,8,11,12]对基于性能和能耗平衡策略的任务调度和资源管理进行了研究。文献[9]提出了基于构件重构的路由器能耗细粒度调整方法,仿真表明其降低了 25% 能耗。文献[8]开发的 Green Flash 系统采用专用硬件和软硬件协同设计的方法提高能效。文献[10]详细研究了虚拟化技术对应用和系统性能的影响,并进行了可用性评测研究。

结束语 文中从构建高效能系统出发,在分析了现有混合集群组织方式 UNNS 和 NNUS 的基础上,提出了一种面向高效能的集群组织方式 3NS,并据此构建了由 10 台通用服务器和共计 20 片 FPGA 组成的原型实验平台。为了清晰描述运行在原型验证平台的应用任务及其运行方法,提出了一种基于主动认知决策的高效能模型。然后,按照该模型的思想实现了视频拷贝检测和 word 2010 口令恢复两个应用,基于构件重构实现了应用结构的动态改变。实验结果表明,该模型能够优化应用结构,不断提高系统效能。在模拟真实互联网流量曲线的负荷模型下,与静态的运行结构相比,基于该模型的系统的效能提高了 58%,并且提供了与静态系统相同的性能。

该模型的本质是一个包含认知、决策和行为系统,并定义它们运行机制的一个框架,它的有效性依赖于认知函数的准确性、决策系统的高效性和行为系统的灵敏性,并且还需要一个合理的结构重构机制,如本文使用的基于构件重构的结构重构机制。这些都是下一步需要研究的方向。

- [1] 洪学海,詹剑锋,樊建平,等. 应用驱动的高效能计算机系统的研究与发展[J]. 计算机研究与发展,2007,44(10):1633-1639
Hong Xue-hai, Zhan Jian-feng, Fan Jian-ping, et al. Research Progress of Application-Driven High Productivity Computing System[J]. Journal of Computer Research and Development, 2007,44(10):1633-1639
- [2] Jones D H, Powell A, Bouganis C, et al. GPU Versus FPGA for High Productivity Computing [C]// 2010 International Conference on Field Programmable Logic and Applications (FPL). Milano: IEEE, 2010: 119-124
- [3] Kornaros G, Pnevmatikatos D. Dynamic Power and Thermal Management of NoC-Based Heterogeneous MPSoCs[J]. ACM Trans. Reconfigurable Technol. Syst., 2014, 7(1): 1-26
- [4] Khan U A, Rinner B. Online learning of timeout policies for dynamic power management[J]. ACM Trans. Embed. Comput. Syst. (TECS), 2014, 13(4): 1-25
- [5] El-Ghazawi T, El-Araby E, Huang M, et al. The promise of high-performance reconfigurable computing[J]. IEEE Computer, 2008, 41(2): 69-76
- [6] Sheikh H F, Tan H, Ahmad I, et al. Energy- and performance-aware scheduling of tasks on parallel and distributed systems [J]. ACM Journal on Emerging Technologies in Computing Systems (JETC), 2012, 8(4): 1-37
- [7] Tsoi K H, Luk W. Axel: a heterogeneous cluster with FPGAs and GPUs[C]// Proceedings of the 18th annual ACM/SIGDA International Symposium on Field Programmable Gate Arrays. California: ACM, 2010: 115-124
- [8] Donofrio D, Oliner L, Shalf J, et al. Energy-Efficient Computing for Extreme-Scale Science[J]. IEEE Computer, 2009, 42(11): 62-71
- [9] 袁博, 汪斌强. 一种基于构件重构的路由器能耗细粒度调整方法[J]. 计算机学报, 2013, 36(7): 1526-1537
Yuan Bo, Wang Bin-qiang. An Energy Meticulous-Grained Scheduling Algorithm for Routers Based on Component Reconfigure [J]. Chinese Journal of Computers, 2013, 36(7): 1526-1537
- [10] Che Jian-hua, He Qin-ming, Ye Ke-jian, et al. Performance combinative evaluation of typical virtual machine monitors[M]// High Performance Computing and Applications. Springer, 2010: 96-101
- [11] Nahapetian A, Brisk P, Ghiasi S, et al. An Approximation Algorithm for Scheduling on Heterogeneous Reconfigurable Resources[J]. ACM Trans. Embed. Comput. Syst., 2009, 9(1): 1-20
- [12] Chao Jing, Zhu Yan-min, Li Ming-lu. Energy-efficient scheduling on multi-FPGA reconfigurable systems[J]. Microprocessors and Microsystems, 2013, 37(6/7): 590-600
- [13] Dong Hui, Huang Le-tian, Wang Jun-shi, et al. Combining Task Scheduling in Power Adaptive Dynamic Reconfigurable System [J]. Journal of Electronic Science and Technology, 2012, 10(4): 296-301
- [14] McIntire D, Stathopoulos T, Reddy S, et al. Energy-Efficient Sensing with the Low Power, Energy Aware Processing (LEAP) Architecture[J]. ACM Trans. Embed. Comput. Syst., 2012, 11(2): 1-36
- [15] 中国互联网络信息中心. 第 25 次中国互联网络发展状况统计报告[R]. 2010
China Internet Network Information Center. 25th Statistical Report of Chinese Internet Development[R]. 2010