

基于异构感知静态调度与动态线程迁移的异构多核调度机制

张 苗,张德贤

(河南工业大学 信息科学与工程学院, 郑州 450001)

(zhangmiao7414@sina.com)

摘 要:异构多核处理器体系结构可以有效减少功耗开销,是处理器发展的趋势,负载不平衡问题会造成处理器执行的不稳定。提出一种基于异构感知的静态调度和动态线程迁移相结合的异构多核调度机制,解决了不同核之间的负载平衡问题,提高了吞吐量。仿真实验通过将此调度机制与静态调度策略(SS)比较,表明该机制提高了异构多核处理器的性能并保证了执行过程的稳定性。

关键词:静态调度;动态线程迁移;异构感知

中图分类号:TP316 **文献标志码:**A

Heterogenous multi-core mechanism based on heterogeneity-aware static scheduling and heterogeneity-aware dynamic thread migration

ZHANG Miao, ZHANG De-xian

(School of Information Science and Engineering, Henan University of Technology, Zhengzhou Henan 450001, China)

Abstract: Heterogeneous multi-core architecture can effectively reduce processor power dissipation, which becomes the development trend for processor. Whereas, the unbalanced load problem of heterogeneous multi-core processor may cause unstable execution of heterogeneous multi-core processor. This paper proposed a mechanism based on Heterogeneity-Aware Static Scheduling (HASS) and Heterogeneity-Aware Dynamic thread Migration (HADDM) to ensure the load balance among various cores. The simulation experiment shows that the mechanism improves the execution performance and ensures the stability of heterogeneous multi-core processor.

Key words: Static Scheduling (SS); Dynamic Thread Migration (DM); Heterogeneity-Aware (HA)

0 引言

采用异构多核处理器的目的是为了将不同的任务配置给不同的处理器,这样不仅可以减少功耗,还可以保证微处理器结构的复杂程度在可控范围之内。异构多核处理器上包含了不同类型的处理核,不同类型的处理核对于相匹配的应用程序有良好的性能。异构多核是可以同时开发指令级并行(Instruction-Level Parallelism, ILP)和线程级并行(Thread-Level Parallelism, TLP)的处理器体系结构,对于包含有大量ILP的串行应用程序,可通过大问题宽度的超标量处理核来处理,多个小问题宽度的处理核可以用来开发TLP。当应用程序在相匹配的处理核上执行时,可以得到很好的性能和较低的功耗,原因是短的执行时间和高的执行效率^[1]。

Balakrishnan 等人^[2]研究发现,处理核的不对称性会对现有的应用产生较大的负面影响,性能也很难预测,而且,由于现在的操作系统不能平衡多个处理核的负载,因此不但不能有效利用异构多核处理器,甚至还会使处理器的执行出现不稳定的现象^[3]。针对此情况,设计了一种非单调性的异构多核处理器体系结构,提出了一种基于异构感知的静态调度(Static Scheduling, SS)和动态迁移相结合的调度机制,解决了不同处理核之间的负载平衡问题,保证了系统的稳定性,同时由于该机制能够同时开发TLP和ILP,所以吞吐量较大,提高了异构多核处理器的执行性能。

1 异构多核调度模型

定义1 异构多核的异构性。是指处理核有不同的执行频率、发射宽度、cache大小和其他的一些硬件特征。

异构多核的优点是,用异构的处理核来处理具有不同类型的应用程序或是同一应用程序的不同部分。所有核心具有的相同的工业标准结构(Industrial Standard Architecture, ISA),使得不同处理核上的线程可以进行无缝迁移。

定义2 异构多核的稳定性。当应用执行时,多个处理核心有相等或相近的执行性能和内部状态。

定义3 多核体系结构的单调性。指多个核心的性能参数具有一定的顺序,并且对所有的应用程序都会保持该顺序。

在异构多核处理器上的多个处理核之间不能存在单调性,其原因在于具有单调性的多核处理器会造成性能的降低和执行的不稳定,并且高性能核的功耗会占总功耗的绝大部分^[4]。不能够感知异构性的任务调度器会给每个处理核都分配相同量的任务,从而导致负载的不平衡,当高性能核的功耗占总功耗一大部分时,多核处理器就显得不稳定。本文设计了一种包含两种处理核类型的异构多核处理器,使得各处理核之间不存在单调性,每一种处理核不再对所有类型的应用程序都有很好的处理性能,而是只适合某一类特殊的应用。

Ekman 等人研究了在相同面积的芯片上如何权衡处理核的个数和每个处理核的发射宽度,研究发现,选择合适的发射宽度会使系统有良好的鲁棒性,并且对单线程和多线程应用

收稿日期:2010-12-31;修回日期:2011-02-07。 基金项目:河南工业大学校基金资助项目(09XJC017)。

作者简介:张苗(1979-),女,河南开封人,讲师,硕士,主要研究方向:模式识别、智能计算; 张德贤(1961-),男,河南新密人,教授,博士,主要研究方向:计算机智能技术。

都有较好的性能^[5]。文献[6]说明了异构多核处理器对多道程序有良好的性能,并证明了异构多核可以开发更宽域的TLP。根据Ekman等人的研究结果,我们选择了发射宽度为8的超标量乱序处理核来执行单线程的应用程序,通过开发ILP来实现高性能;选择发射宽度为4的简单核来执行多线程的应用程序,从而开发TLP。

基于异构感知的静态调度和动态迁移相结合的异构多核

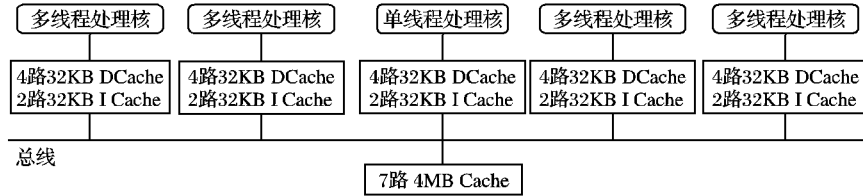


图1 异构多核设计方案

2 基于 HASS 和 HADM 的异构多核调度

2.1 异构感知的静态调度

因为处理器包含了两种类型的处理核,因此,必须开发一种有效的调度机制把线程分配到合适的处理核上,分配线程的依据是线程类型、处理核类型和负载程度。

基于异构感知的静态调度和动态迁移相结合的异构多核调度器可以识别出线程的类型,并且能够感知到多核的异构性,并在此基础上实现线程和处理核的优化匹配。在调度器中,把处理核定义为四元组 $C = (C_{type}, C_{number}, C_{ILP}, C_{load})$ 。 C_{type} 代表处理核的类型, C_{number} 代表编号, C_{ILP} 代表当前执行的 ILP 的并行度, C_{load} 代表计算任务的负载值。 C_{type} 和 C_{number} 是指导静态调度的静态信息, C_{ILP} 和 C_{load} 是指导动态迁移的动态信息,均可从任务执行的采样中获得。调度器定义了一个等待线程队列 $T_w[N]$,用来保存等待线程的各种信息,还定义了处理核队列 $C[M]$,用来保存所有处理核的信息。

异构感知的静态调度策略 (Heterogeneity-Aware Static Scheduling, HASS) 首先由调度器识别出线程 $T_w[1]$ 的类型,如果在处理核队列 $C[M]$ 中有一个闲置且适合线程 $T_w[1]$ 的处理核,则该线程就会立即被映射到这个处理核;如果不是合适的处理核或是当前处理核忙碌,则会分析下一个线程 $T_w[2]$;如果线程队列 $T_w[N]$ 不满时,新到达的线程则存放在队列的尾部。具体算法如算法 1。

算法 1 异构多核静态调度策略。

- 1) 初始化等待线程队列 $T_w[N]$;
- 2) 初始化处理核队列 $C[M]$;
- 3) for each $T_w[i]$
- 4) if ($C[j].type == T_w[i].type$ and $C[j].load == light$)
- 5) scheduling $T_w[i]$ to $C[j]$;
- 6) else
- 7) $T_w[i].state = waiting$;
- 8) end if
- 9) if ($T_w[] = full$)
- 10) $T_w.bottom = new\ thread$
- 11) end if
- 12) end for

2.2 异构感知的动态迁移

应用程序在执行时有不同的运行阶段,因此,随着应用程序运行阶段的转换和处理核执行状态的迁移,同一个任务最匹配的处理核也会变化。在这个过程中,调度器应该能够感知并触发一个调优机制,对应用任务进行动态的迁移。针对此情况,提出了异构感知的动态迁移策略 (Heterogeneity-

调度机制的处理器包含两种不同类型的处理核:单线程处理核有 1 个,主频为 2.4 GHz;多线程处理核有 4 个,主频为 1.8 GHz。单线程处理核的指令 Cache 为 2 路组结构,数据 Cache 为 4 路组结构,二者大小均为 64 KB;多线程处理核的指令 Cache 为 2 路组结构,数据 Cache 为 4 路组结构,二者大小均为 32 KB,两类处理核共享了一个片上 7 路组结构的 Cache,大小为 4 MB。如图 1 所示。

Aware Dynamic Migration, HADM), 该策略包含两类触发机制,第一类是线程在不同类型处理核之间的迁移;第二类是线程在同一类型处理核之间的迁移。以上两类触发机制触发的依据分别是 ILP 度和负载量, ILP 度和处理核负载可以从硬件性能计数器中获得。调度器还定义了一个运行线程队列 $T_r[N]$,包含了正在运行的线程的动态信息。

在程序运行时, HADM 第一类机制的触发必须满足三个条件:1) 当程序运行到新的阶段时,两个阶段的 ILP 值改变超过 100%;2) 新的运行阶段必须持续 1 亿个时钟周期;3) 存在一个处理核是空载或是轻载。

第二类机制的触发必须满足两个条件:1) 存在一个重载的处理核;2) 存在一个同类型的空载或轻载处理核。通过两类触发机制的动态调整,达到改善应用程序执行稳定性的目的。HADM 的运行过程如算法 2。

算法 2 异构多核动态迁移策略。

- 1) 初始化 $T_r[N]$;
- 2) 初始化 $C[M]$;
- 3) for each $T_r[i]$ on type2 core{
- 4) if ($Ctype2[k].T_r[i].newILP == 2 * Ctype2[k].T_r[i].oldILP$ and $T_r[i].newstate == 100\ million\ cycles$ and $Ctype1.load == light$)
- 5) migrating ($T_r[i]$);
- 6) end if
- 7) if ($Ctype2[p].load == heavy$ and $Ctype2[q].load == light$)
- 8) migrating ($T_r[i]$)
- 9) end if
- 10) end for
- 11) for each $T_r[i]$ on type1 core
- 12) if ($Ctype1.T_r[i].newILP == Ctype1.T_r[i].oldILP/2$ and $T_r[i].newstate == 100\ million\ cycles$ and $Ctype2[j].load == light$)
- 13) migrating ($T_r[i]$);
- 14) end if
- 15) end for

3 仿真实验结果及分析

仿真实验使用模拟器 SimWatch^[7]来模拟异构多核处理器。SimWatch 整合了模拟器 Simics 和 Watch,可以对多核处理器进行建模,并模拟处理器周期级的性能和功耗。

测试程序(如表 1)是从基准测试集 SPEC 2000 Suite 中选择的,与实际的应用程序略有不同,主要是为了保证测试程序在不同平台间的移植性和比较性,且 SPEC 主要是用来测试 CPU 性能,强调反映真实应用(如实际负载等)的执行情况。

仿真实验通过对测试程序运行过程的采样,把采样信息发给调度器,使得调度器决定是否触发或什么时候触发动态线程迁移策略。

表1 测试程序

测试程序名称	类型	语言	描述
256. bzip2	整型	C	数据压缩
181. mcf	整型	C	组合优化
186. crafty	整型	C	棋类游戏
252. eon	整型	C++	计算可视化
300. twolf	整型	C	布局与路由仿真
177. mesa	浮点型	C	3D 图形库
179. art	浮点型	C	自适应推理/神经网络模拟
168. wupwise	浮点型	F77	动力学/量子论
172. mgrid	浮点型	F77	多网格法求解 3D 势场
188. ammp	浮点型	C	计算化学

为了验证 HASS + HADM 的有效性,仿真实验将基于异构感知的静态调度和动态迁移相结合的异构多核调度策略与静态调度策略^[5]进行了比较,并采用吞吐量作为系统稳定性的评价指标,采用加权加速比作为处理器性能的评价指标。

SS 策略不能够感知异构性,只要有可用的处理核,线程就会被调度。我们使用三种策略对基准程序各执行了 3 次,各调度算法的规范化吞吐量如图 2 所示。通过实验结果可以看出,SS 的性能变化很大,HASS 的性能比较稳定,基于 HASS + HADM 相结合的调度策略有效保证了多个处理核之间的负载均衡,有最好的稳定性。

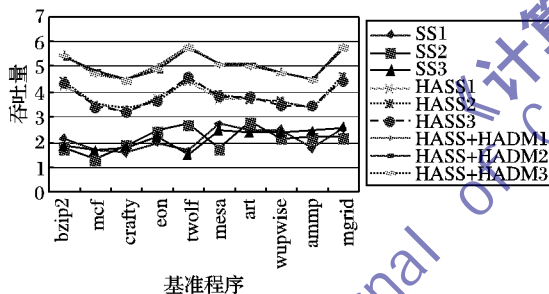


图2 基准程序标准化吞吐量结果

加权加速比的计算方法是,采用简单调度策略应用任务执行多次的时间除以采用异构感知的调度策略执行多次的时间。不同基准测试程序的加权加速比如图 3 所示。实验结果显示了 HASS 下的执行性能要优于 SS,平均加权加速比为 1.8,HASS + HADM/SS 的平均加权加速比达到了 2.4。

4 结语

在异构多核处理器中,采用不同的任务分配与调度算法

会导致不同的执行效率,任务分配的目的就是找出一个任务组合,使它们执行效率最高^[8]。本文首先阐述了异构多核处理器的优缺点,然后设计了一个处理核之间不存在单调性的异构多核处理器体系结构,并提出了一种基于异构感知的静态调度策略和动态迁移相结合的调度策略,解决了多个处理核之间负载不平衡的问题,充分发挥了异构多核的异构性优势。仿真实验表明,基于异构感知静态调度和动态迁移相结合的调度机制能同时开发 TLP 和 ILP,有效减少平均任务执行时间,提高资源利用率,具有较好的稳定性。

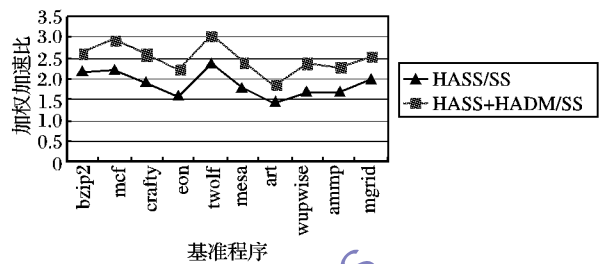


图3 加权加速比

参考文献:

- [1] 彭蔓蔓, 董立超, 王颖. 异构多核处理器的任务分配及能耗的研究[J]. 计算机应用研究, 2010, 27(5): 1719-1731.
- [2] BALAKRISHNAN S, RAJWAR R, UPTON M, et al. The impact of performance asymmetry in emerging multi-core architectures[C]// Proceedings of International Symposium on Computer Architecture. Washington, DC: IEEE Computer Society, 2005: 506-517.
- [3] KUMAR R, FARKAS K I, JOUPPI N P, et al. Single-SA heterogeneous multi-core architectures: The potential for processor power reduction[EB/OL]. [2010-10-20]. <http://www.microarch.org/micro36/html/pdf/kumar-SingleSAHeterogen.pdf>.
- [4] KUMAR R, TULLSEN D M, NORMAN P. Jouppi: Core architecture optimization for heterogeneous chip multiprocessors[C]// Proceedings of the 15th International Conference on Parallel Architectures and Compilation Techniques. New York: ACM, 2006: 23-32.
- [5] EKMAN M, STENSTROM P. Performance and power impact of issue-width in chip-multiprocessor cores[C]// International Conference on Parallel Processing. New York: IEEE, 2003: 359-368.
- [6] KUMAR R, TULLSEN D M, RANGANATHAN P, et al. Single-ISA heterogeneous multi-core architectures for multithreaded workload performance[C]// International Symposium on Computer Architecture. Washington, DC: IEEE Computer Society, 2004: 2173-2179.
- [7] CHEN J, DUBOIS M, STENSTROM P. SimWattch: An approach to integrate complete-system with user-level performance/power simulators[J]. IEEE Micro, 2003, 27(4): 34-48.
- [8] 蒋建春, 汪同庆. 异构多核处理器的任务调度算法[J]. 计算机工程与应用, 2009, 45(33): 52-56.

(上接第1780页)

- [5] LI CHUHAO, SUN YONGHE, JIA YANHUI, et al. An improved ranking approach to AHP alternatives based on variable weights[C]// Proceedings of IEEE the 7th World Congress on Control and Automation. New York: IEEE, 2008: 8255-8260.
- [6] 李玲娟, 汤文宇, 王汝传. 一种适用于 CBR 的特征权值计算方法[J]. 西安邮电学院学报, 2007, 12(5): 58-61.
- [7] JIN TAO, SHEN HUIZHANG. Feature selection and weighting method based on similarity rough set for CBR[C]// IEEE International Conference Service Operations and Logistics, and

Informatics. New York: IEEE, 2006: 948-952.

- [9] 杨健, 杨晓光等. 一种基于 k-NN 的案例相似度权重调整算法[J]. 计算机工程与应用, 2007, 43(23): 8-11.
- [10] 岳超源. 决策理论与方法[M]. 北京: 科学出版社, 2003.
- [11] 张文修. 粗糙集理论与方法[M]. 北京: 科学出版社, 2006.
- [12] 周平, 丁进良, 岳恒, 等. 基于相似粗糙集的案例特征权值确定新方法[J]. 信息与控制, 2006, 35(3): 229-334.
- [13] 苗夺谦. Rough Set 理论中连续属性的离散化方法[J]. 自动化学报, 2001, 27(3): 296-30.