

# 基于 CMP 系统的并行编程模式研究

胥秀峰, 鲍广宇, 黄海燕, 吴亚宁

(解放军理工大学 指挥信息系统学院, 江苏 南京 210007)

**摘要:**研究基于 CMP (Chip Multiple Processors, 片上多处理器) 系统的并行编程模式旨在建立开发 CMP 系统上并行程序的整套方法。首先简要介绍了多核并行计算, 然后通过对 CMP 系统上并行计算问题的综合归纳, 提出了基于 CMP 系统的并行编程模式的概念模型, 这个概念模型包含并行体系结构、并行算法设计模型、开发环境、并行程序实现模型四个核心要素; 其次, 对各并行编程模式各要素及其子概念的内涵进行了阐释; 最后以实例对并行编程模式进行说明, 初步验证了这套编程模式的合理性。

**关键词:** CMP 系统; 并行计算; 编程模式

中图分类号: TP302.1

文献标识码: A

文章编号: 1673-629X(2014)07-0080-04

doi: 10.3969/j.issn.1673-629X.2014.07.020

## Study on Patterns for Parallel Programming Based on CMP System

XU Xiu-feng, BAO Guang-yu, HUANG Hai-yan, WU Ya-ning

(Institute of Command Information System, PLAUST, Nanjing 210007, China)

**Abstract:** Studying patterns for parallel programming based on CMP system is designed to build and develop the whole way for parallel program on CMP system. Firstly, briefly introduce the multi-core parallel computing, then put forward a conceptual model of patterns for parallel programming based on CMP system by summarizing the problem of parallel computing, and the patterns contains four elements of parallel architecture, parallel algorithm design model, development environment, parallel program implementation model. Then, describe the main connotation of the concepts. Finally illustrate the parallel programming patterns by a example, initially verifying the reasonableness of the patterns.

**Key words:** CMP system; parallel computing; patterns for programming

## 0 引言

各型 CMP 系统已应用于军事领域, 例如多核的 PC、智能手机、平板电脑等。这些 CMP 平台, 有的已经作为指挥信息系统的硬件组成(如 PC), 还有的未来可能被运用到指挥信息系统中(如智能手机)。但当前多核平台上的并行软件开发技术还远没有普及化、大众化, 各类多核系统运行的主要还是传统串行程序。因此, 无论是从提高程序效率的角度还是从扩展应用形式的角度来看, 研究多核并行计算, 开发适用于各类数量众多的 CMP 系统的并行应用程序, 有很重要的现实意义。

## 1 多核并行计算简介

并行计算(parallel computing)已有几十年的发展历史, 研究者包括计算数学家、计算机科学家、软件工程师和应用领域专家, 产生了丰富的研究成果<sup>[1]</sup>。相对于串行计算, 并行计算是在并行计算机上执行的计算。这里, 并行计算机指拥有多个计算资源的计算机, 多个计算资源可以是单个多核 CPU、通过网络连接的多个 CPU 或以上两者的组合<sup>[2]</sup>。

并行计算的研究内容主要可归纳为四大部分: 并行计算的硬件平台、并行计算的理论基础、并行计算的软件支撑和并行应用<sup>[1]</sup>。多核并行计算是并行计算在多核计算机(即采用多核 CPU 的计算机)出现后自然

收稿日期: 2013-09-12

修回日期: 2013-12-23

网络出版时间: 2014-04-24

基金项目: 国家自然科学基金资助项目(61174198)

作者简介: 胥秀峰(1989-), 男, 硕士研究生, 研究方向为系统仿真与评估; 鲍广宇, 博士, 硕士生导师, 教授, 研究方向为系统仿真、作战信息管理。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20140424.0818.059.html>

派生出的新领域,其研究的核心内容继承于一般并行计算,也大致包括以上四个方面。并行编程模式的研究主要涉及后三个部分。

## 2 基于 CMP 系统的并行编程模式

### 2.1 基于 CMP 系统的并行编程模式概念模型

多核处理器的普及使开发并行程序不再局限于高性能计算领域,但由于并行计算机的体系结构及解决具体问题的并行算法、并行程序,都不同于传统串行体系,所以开发并行程序不能直接运用开发串程序的整套成熟方法。在此背景下,文中提出了一种基于 CMP 系统的并行编程模式,试图建立一套覆盖体系结构分析、并行算法设计、并行程序实现的方法框架,支撑 CMP 系统上的并行程序的开发。

基于 CMP 系统的并行编程模式(以下称并行编程模式)是一个方法框架,用于在 CMP 系统上开发并行程序。并行编程模式可分为概念模型与过程模型两个视图,文中主要论述并行编程模式的概念模型。

并行编程模式的概念模型限定了并行编程模式的边界及内涵,其主要概念可用四元组表示:

并行编程模式(Patterns for Multi-core Parallel Programming)记为 PMPP,则有:

$$\text{PMPP} = (\text{PA}, \text{PAD}, \text{DE}, \text{PM})$$

其中,PA 指并行体系结构(Parallel Architecture);PAD 指并行算法设计模型(Parallel Algorithm Design model),PAD = (PCM, DM),其中 PCM(Parallel Computation Model)为并行计算模型,DM(Design Model)为设计方法模型;DE 指开发环境(Development Environment),DE = {OS, PE},其中 OS(Operating System)为操作系统,PE(Program Environment)为编程环境;PM 指并行程序实现模型(Parallel Program Implementation Model),PM = {PPM, IM},其中 PPM(Parallel Programming Model)为并行编程模型,IM(Implementation Model)指实现方法模型<sup>[3-8]</sup>。

并行编程模式的主要作用是设计并行算法与实现并行程序,所以在四个基本概念中并行算法设计模型与并行实现模型处于核心地位。

### 2.2 并行体系结构

在并行编程模式中,并行体系结构是指并行计算机硬件的概念结构与功能特性,是对并行计算机中存储器的互联网络及其作用与通信方式,或其他体系结构特征的描述<sup>[9-11]</sup>。

并行机系统一般可分为 6 类:单指令多数据流(Single-Instructing Multiple-Data, SIMD),并行向量处理机(Parallel Vector Processor, PVP),对称多处理机(Symmetric Multi Processor, SMP),大规模并行处理机

(Massively Parallel Processor, MPP),工作站机群(Cluster Of Workstations, COW)和分布共享存储(Distributed Shared Memory, DSM)<sup>[1]</sup>。

### 2.3 并行算法设计模型

#### 2.3.1 并行计算模型

并行计算模型是算法设计者与体系结构之间的桥梁,它包括机器参数(也称为模型参数)、计算行为和成本函数三个要素。并行计算模型的机器参数是从并行机中抽取的若干个能反映计算特性、可计算或可测量的参数;以机器参数为自变量,按照模型定义的计算行为可构造成本函数(Cost Function),根据成本函数能够分析具体并行算法的时空复杂度<sup>[8]</sup>。

并行计算模型主要有共享存储模型、分布存储并行计算模型、层次存储并行计算模型,此外还有一些其他适用范围相对较窄的并行计算模型。

#### 2.3.2 设计方法模型

并行算法设计模式中的设计方法模型是设计并行算法的过程模型,它以并行计算模型为支撑,可分为五个步骤:

1)并行性分析。对问题进行分析,总结出问题的并行特性。

2)划分并行任务。也称并行任务分解,即将原计算问题分割成一些小的计算任务,以充分挖掘并行执行的机会。划分的方式有数据划分与功能划分两种。

3)协调任务关系。划分并行任务工作完成后,原问题被分解成多个子任务,必须规定划分所得的各子任务之间的关系。子任务的关系有数据依赖,顺序依赖,互不相关(没有数据依赖,没有顺序依赖)几种。

4)设计任务结构。设计任务结构指设计各子任务的内部细节。

5)并行任务归并。并行的子任务完成计算后必须把各结果进行整合,最后得到原问题的结果。把子任务结果进行整合的过程就是任务归并。

### 2.4 开发环境

并行编程模型中,开发环境是指开发并行软件应用所需的各类支撑软件,主要包括操作系统、编程环境。操作系统的进程/线程管理机制决定了 CMP 系统上并行程序的执行方式,编程环境包括编译器、编程语言、开发工具等,编程环境可根据需要灵活构建<sup>[12]</sup>。

例如,在 CMP 平台开发基于 NeoKylin 操作系统的应用,除操作系统外,编程环境可配置为:

开发平台:Code::Blocks 集成开发环境;

编译器:GCC 4.4.5(集成 OpenMP 的 API)。

### 2.5 并行程序实现模型

#### 2.5.1 并行编程模型

并行编程模型是将算法转化为程序的工具,它的

主要内容是定义基本并程序结构(程序的并行执行模式)或提供并行程序模板等,并行编程模型的形式可以为传统编程语言基础上的编程框架或 API,也可以由一种专门的编程语言承载。例如,OpenMP 就是一种并行编程模型,它是对传统编程语言的扩展,其并程序结构为多线程的“FORK-JOIN”(派生-合并)模式,OpenMP 目前有 Fortran、C/C++ 等语言的 API (API 包括若干编译指令、运行库和环境变量)<sup>[3,13]</sup>。其他的并行编程模型还有很多,例如当前应用比较广泛的 MapReduce。

2.5.2 实现方法模型

实现方法模型是编写并程序的过程模型,它依赖并行编程模型,可分为建立开发环境、设计数据结构、设计程序结构、设计协同动作、编写程序代码、测试六个步骤,实现方法模型的流程图如图 1 所示。

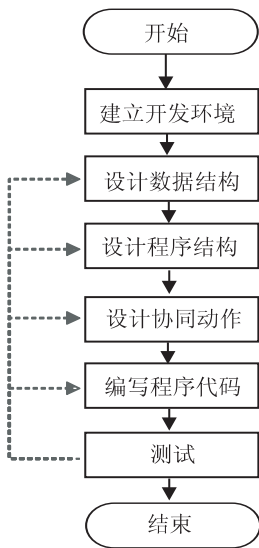


图 1 实现方法模型流程图

1)建立开发环境。由构建操作系统环境与构建编程环境两部分组成,由支撑语言、软件工具和应用软件包等组成。

2)设计数据结构。数据建立什么样的逻辑结构是在设计算法时考虑的问题,在程序设计阶段所谓设计数据结构主要考虑的是采用何种存储结构。

3)设计程序结构。CMP 系统上,算法中原问题经过划分后得到的各子任务,在程序中对应在各 CPU 内核上执行的线程,设计程序结构就是设计这些线程内部的执行序列。

4)设计协同动作。CMP 系统上,并程序采用多线程的最终是为了共同完成统一计算任务,这些线程必然需要进行协同。线程的协同主要指同步与通信,同步是指线程之间有顺序关系,通信是指线程之间有数据交换。

5)编写程序代码。完成上述的设计过程后,进行代码的编写。

6)测试。测试并行程序的正确性及性能<sup>[14]</sup>。

3 实例

以求圆周率  $\pi$  的简单例子说明基于 NeoKylin 的 CMP 系统并行编程模式。 $\pi$  值的计算方法有多种,该例采用数值积分法,计算公式如下( $N$  越大,结果近似度越高):

$$\pi = \int_0^1 \frac{4}{1+x^2} dx \approx \sum_{0 \leq i \leq N} \frac{4}{1 + (\frac{i+0.5}{N})^2} \cdot \frac{1}{N} \quad (1)$$

运用基于 NeoKylin 的 CMP 系统并行编程模式,从分析硬件结构、设计并行算法到编写求  $\pi$  值的程序过程如表 1 所示。

表 1 求  $\pi$  的程序并行编程模式

阶段	步骤	内容
体系结构分析	确定结构类型	CMP 系统为 MIMD 系统
	分析存储系统	CMP 系统采用多级存储结构,各 CPU 内核共享存储器
	分析通信方式	可采用消息传值与共享变量两种方式通信
	并行计算模型	CMP 系统可用 APARM、QSM 模型进行抽象
并行算法设计	并行性分析	公式(1)是算法设计的主要依据。公式(1)的计算是一个累加和的过程;各累加项的值分别计算,不依赖其他项的结果;各累加项值的总和为 $\pi$ 的最终结果
	划分并行任务	由问题的并行特性,可把 $N$ 个累加项平均划分到各 CPU 内核上进行计算
	协调任务关系	划分在 $m$ 个 CPU 内核上的各子任务计算的都是公式(1)的不同累加项,各子任务间没有依赖,为平行关系,可以同时进行计算且互不影响
	设计任务结构	问题比较简单, $m$ 个 CPU 内核上的子任务就是分别求 $N/m$ 个累加项的累加和
	并行任务归并	各子任务求得的结果直接累加即得到原问题的结果
	算法性能评估	用 APARAM 或 QSM 计算模型分析,上述以公式(1)求 $\pi$ 的并行算法较公式(1)的串行算法时间复杂度更低(因并行、串行算法均是公式(1)得出,两者有相似性,并行算法可视为串行算法的改进),并行算法执行时间近似为串行算法执行时间的 1/4

(续表)

阶段	步骤	内容
并行程序实现	建立开发环境	操作系统:NeoKylin 桌面操作系统;开发平台:Code::Blocks 集成开发环境;编译器 GCC 4.4.5;并行编程模型:OpenMP(已集成到 GCC 中)
	设计数据结构	问题不涉及大量的数据,不需要专门的数据结构,用 OpenMP 模型实现算法时,只需几个变量就可以进行各累加项的计算
	设计程序结构	因为算法主要是累加计算,所以程序主要结构是循环,即 $m$ 个 CPU 内核上分别进行 $N/m$ 次循环,循环主要是进行计算各累加项,并把结果进行累加
	设计协同动作	OpenMP 用多线程方式把计算分配到各 CPU 内核上,算法中各子任务为平行关系,相应各 CPU 内核上的线程也没有依赖关系,因此可异步独立运行,不必考虑同步
	编写程序代码	编码实现程序,主要代码:(略)
	正确性测试	输入若干不同的典型 $N$ 值,证明求得的 $\pi$ 符合要求
	性能测试	公式(1)求 $\pi$ 的串行程序所用计算时间近似于并行改进程序计算时间的 4 倍

4 结束语

文中对 CMP 系统上的并程序开发进行了研究,提出了 CMP 系统并行编程模式的概念模型,并给出实例说明,为 CMP 系统上编写并行程序提供了一种框架。并行编程模式的过程模型以及概念模型中各元素的结构细节,将在以后的工作中继续研究。

参考文献:

[1] 陈国良. 并行计算-结构·算法·编程[M]. 修订版. 北京: 高等教育出版社,2003.

[2] Barney B. Introduction to parallel computing[EB/OL]. 2012. <https://computing.lnl.gov>.

[3] 董仁举. 基于 CMP 集群的并行编程模型的研究与应用[D]. 曲阜:曲阜师范大学,2011.

[4] Momcilovic S,Sousa L. Modeling and evaluating non-shared Memory CELL/BE type multi-core architectures for local image and video processing[J]. Journal of Signal Processing Systems,2011,62(3):301-318.

[5] 陈小兰. Linux 应用程序多核并行化方法研究与实现[D]. 成都:西南交通大学,2010.

[6] 宁林志. 基于资源优化的 CMP 体系结构研究[D]. 哈尔

滨:哈尔滨工程大学,2008.

[7] Chen Guoliang,Sun Guangzhong,Zhang Yunquan,et al. Study on parallel computing[J]. Journal of Computer Science and Technology,2006,21(5):665-673.

[8] Mattson T G,Sanders B A,Massingill B L. Patterns for parallel programming[M]. USA:Addison Wesley/Pearson,2005.

[9] 奚海波. 并行编程技术在多核处理器上的研究与应用[D]. 长春:东北师范大学,2010.

[10] 伊君翰. 基于多核的并行编程模型[D]. 上海:复旦大学,2008.

[11] Jeff P,John D,Bill G. From single core to multi-core:preparing for a new exponential[C]//Proc of IEEE international conference on computer-aided design. Washington: IEEE Computer Society,2006:67-72.

[12] 曹 皓. 多核处理器体系结构下 Linux 调度机制的研究[D]. 呼和浩特:内蒙古大学,2011.

[13] Kil J K. Parallel quick sort algorithms analysis using OpenMP 3.0 in embedded system[C]//Proc of 11th international conference on control,automation and systems. [s. l.]:[s. n. ], 2011:757-761.

[14] Saifullah A, Li J, Agrawal K, et al. Multi-core real-time scheduling for generalized parallel task models[J]. Real-time Systems,2013,49(4):404-404.

+++++

(上接第 79 页)

报,2001,23(10):889-993.

[9] Soon I Y,Koh S N,Yeo C K. Noisy speech enhancement using discrete cosine transform[J]. Speech Communication,1998,24(3):249-257.

[10] 王 媛,刘 珩. 高噪声环境下基于倒谱距离的语音端点检测算法的实现[J]. 中国农业大学学报,2006,11(2):82-84.

[11] 于迎霞,史家茂. 一种改进的基于倒谱特征的带噪端点检测方法[J]. 计算机工程,2004,30(19):85-87.

[12] 胡光锐,韦晓东. 基于倒谱特征的带噪语音端点检测[J].

电子学报,2000,28(10):95-97.

[13] Beritelli F,Casale S,Cavallaro A. A robust voice activity detector for wireless communications using soft computing[J]. IEEE Journal on Selected Areas in Communication,1998,16(9):1818-1829.

[14] Chang J H. Warped discrete cosine transform-based noisy speech enhancement[J]. IEEE Trans on Circuits and Systems-II:Express Briefs,2005,52(9):535-539.

[15] Seneff S. Real-time harmonic pitch detector[J]. IEEE Transactions on Acoustics,Speech and Signal Processing,1978,26(4):358-365.

作者：[胥秀峰](#)，[鲍广宇](#)，[黄海燕](#)，[吴亚宁](#)，[XU Xiu-feng](#)，[BAO Guang-yu](#)，[HUANG Hai-yan](#)，[WU Ya-ning](#)  
作者单位：[解放军理工大学 指挥信息系统学院](#), 江苏 南京, 210007  
刊名：[计算机技术与发展](#)   
英文刊名：[Computer Technology and Development](#)  
年，卷(期)：2014(7)

参考文献(14条)

1. [陈国良](#) [并行计算-结构@算法@编程](#) 2003  
2. [Barney B](#) [Introduction to parallel computing](#) 2012  
3. [董仁举](#) [基于 CMP 集群的并行编程模型的研究与应用](#) 2011  
4. [Momcilovic S;Sousa L](#) [Modeling and evaluating non-shared Memory CELL/BE type multi-core architectures for local im-age and video processing](#) 2011(03)  
5. [陈小兰](#) [Linux应用程序多核并行化方法研究与实现](#) 2010  
6. [宁林志](#) [基于资源优化的 CMP体系结构研究](#) 2008  
7. [Chen Guoliang;Sun Guangzhong;Zhang Yunquan](#) [Study on parallel computing](#) 2006(05)  
8. [Mattson T G;Sanders B A;Massingill B L](#) [Patterns for parallel programming](#) 2005  
9. [奚海波](#) [并行编程技术在多核处理器上的研究与应用](#) 2010  
10. [伊君翰](#) [基于多核的并行编程模型](#) 2008  
11. [Jeff P;John D;Bill G](#) [From single core to multi-core:prepar-ing for a new exponential](#) 2006  
12. [曹皓](#) [多核处理器体系结构下 Linux 调度机制的研究](#) 2011  
13. [Kil J K](#) [Parallel quick sort algorithms analysis using OpenMP 3. 0 in embedded system](#) 2011  
14. [Saifullah A;Li J;Agrawal K](#) [Multi-core real-time scheduling for generalized parallel task models](#) 2013(04)

引用本文格式：[胥秀峰](#).[鲍广宇](#).[黄海燕](#).[吴亚宁](#).[XU Xiu-feng](#).[BAO Guang-yu](#).[HUANG Hai-yan](#).[WU Ya-ning](#) [基于 CMP系统的并行编程模式研究](#)[期刊论文]-[计算机技术与发展](#) 2014(7)