

**研究生学位论文中期报告**

**报告题目 面向高通量处理器的Benchmark研究**

**学生姓名 苗福涛 学号 201228013229039**

**指导教师 张志敏 职称 研究员**

**学位类别 工学硕士**

**学科专业 计算机系统结构**

**研究方向 处理器结构**

**培养单位 计算技术研究所**

**填表日期 2015.4**

**中国科学院大学制**

**填 表 说 明**

1. 本表内容须真实、完整、准确。
2. “学位类别”名称填写：哲学博士、教育学博士、理学博士、工学博士、农学博士、医学博士、管理学博士，哲学硕士、经济学硕士、法学硕士、教育学硕士、文学硕士、理学硕士、工学硕士、农学硕士、医学硕士、管理学硕士等。
3. “学科专业”名称填写： “二级学科”全称。

**报告提纲**

一、学位论文进展情况，存在的问题，已取得阶段性成果

二、下一步工作计划和内容，预计答辩时间

三、已取得科研成果列表（已发表、待发表学术论文、专利等）

目录

[1 学位论文进展情况 5](#_Toc416982162)

[1.1 高通量应用分类与分析 5](#_Toc416982163)

[1.1.1 基于高通量需求特点的高通量应用分类模型 5](#_Toc416982164)

[1.1.2 基于分类模型的高通量应用分类与Workload提取 6](#_Toc416982165)

[1.1.3 高通量应用典型程序特征总结 8](#_Toc416982166)

[1.2 面向高通量处理器的Benchmark的实现 10](#_Toc416982167)

[1.2.1 提出基于线程的作业处理节点并行化模型 12](#_Toc416982168)

[1.2.2 数据处理类高通量应用Benchmark实现 12](#_Toc416982169)

[1.2.3 数据服务类高通量应用Benchmark实现 14](#_Toc416982170)

[1.2.4 实时交互类高通量应用Benchmark实现 16](#_Toc416982171)

[1.3 实验评估 18](#_Toc416982172)

[1.3.1 对Benchmark程序特性进行了实验评估 18](#_Toc416982173)

[1.3.2 对Tilera TILE-Gx处理和Intel Xeon处理器进行评估 24](#_Toc416982174)

[2 下一步工作计划和内容，预计答辩时间 27](#_Toc416982175)

[3 科研项目参与情况 27](#_Toc416982176)

[4 课程学习情况 28](#_Toc416982177)

# 学位论文进展情况

学位论文所涉及到的工作已基本完成，论文初稿已经完成。

## 高通量应用分类与分析

这一部分工作完成了基于对数据中心典型高通量应用的分类，Workload提取和程序特征分析等几方面的工作。

### 基于高通量需求特点的高通量应用分类模型

本文总结归纳出高通量需求可以分为三种：

1. 单位时间内能够处理尽量大的数据量。
2. 单位时间内能够处理尽量多的请求数。
3. 能够同时支持尽量多的用户在线实时处理数据。

根据这些不同的高通量需求，可以将高通量应用分为三类：数据处理类、数据服务类、实时交互类。

基于此定义基于高通量需求的高通量应用分类模型如图所示：

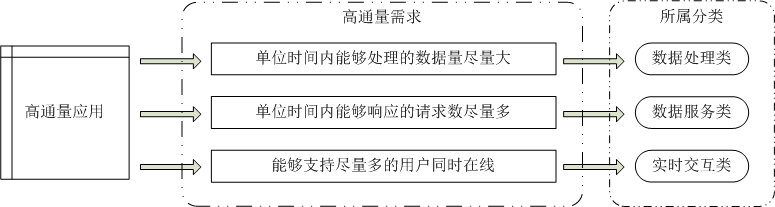


图1基于高通量需求的分类模型

高通量应用具有不同的类型，不同类型的高通量应用作业的定义和高通量需求关注指标是不同的，因而，本文对不同的高通量应用类型进行了：

1. 作业的定义。
2. 性能指标的定义。

三类高通量应用的相关定义分别为：

1）数据处理类

作业：对大数据集切分后的一部分数据进行处理的过程。

指标：一定时间内能够处理的数据量，即对数据的吞吐量。

2）数据服务类

作业：对一个请求进行响应的过程。

指标：一定时间内能够及时处理和响应的请求数量，即对请求响应的吞吐量。

3）实时交互类

作业：维持一个用户的在线状态并处理此用户的数据。

指标：能够支持同时处于链接状态并保证服务实时性的用户数，即对用户接入的吞吐量。

### 基于分类模型的高通量应用分类与Workload提取

分析各典型高通量应用的高通量需求特点，基于分类模型对各个典型的高通量应用进行归类，并提取各个应用中的典型Workload。

得到研究结果如下表所示。

表 1 典型高通量应用Workload提取与划分结果

|  |  |  |  |
| --- | --- | --- | --- |
| 类型 | 应用 | Workload提取与划分 | |
| 数据处理类 | Big Data Analytics | Basic operation | sort |
| grep |
| wordcount |
| Classification | Naïve Bayes |
| SVM |
| Cluster | K-means |
| Fuzzy k-means |
| Segmentation | HMM |
| regression analysis | regression analysis |
| Machine Learning | 半监督学习 | 基于混合产生式模型 |
| 基于低密度划分 |
| 基于图的 |
| 基于不一致性的 |
| 协同训练算法 |
| 集成学习 | 顺序的集成学习方法（Boosting） |
| 并行的集成学习方法（Bagging） |
| 概率图模型 | 贝叶斯网络 |
| 马尔科夫网络 |
| 隐马尔可夫网络模型 |
| 迁移学习 | 归纳迁移学习 |
| 直推迁移学习 |
| 无监督迁移学习 |
| 数据服务类 | Web Search | 网络爬虫 | 广度优先遍历算法BFS |
| 深度优先遍历算法DFS |
| 调度系统（管理优先级排序） |
| Hash Table |
| 建立索引库 | Html parse |
| Directory iterative |
| Index |
| PageRank |
| TF-IDF |
| 响应用户请求 | Bool operations |
| Query parse |
| Database query |
| Bool operation |
| Term weight |
| Document weight |
| MSet |
| Social Network | 社交网络影响力最大化 | Diffusion Degree算法 |
| Greedy算法 |
| 推荐算法 | 协同过滤推荐 |
| 基于内容推荐 |
| 数据抽样方法 | 广度优先抽样法 |
| 点边抽样法 |
| 用户均匀抽样法 |
| 同伴推动抽样法 |
| 随机行走抽样法 |
| 拓扑分析 | 计算平均路径长度 |
| 计算聚集系数 |
| 数据分析 | 基于链接的结构分析 |
| 基于内容的分析 |
| E-commerce | 搜索系统 | 根据不同的关键字建立索引(Index) |
| 响应用户查询(Query) |
| 订单系统 | 订单建立(Create) |
| 订单查询(Select) |
| 订单修改(Update) |
| 状态机(Finite State Machine) |
| 数据库系统 | 商品管理系统 |
| 会员管理系统 |
| 订单管理系统 |
| 推荐系统 | 协同过滤算法 |
| 基于内容的推荐算法 |
| 基于聚类算法的推荐 |
| 基于产品到产品的推荐 |
| 实时交互类 | Radio Network Controller | MACD层 | SDU receive |
| Schedule |
| RLC层 | Entity query |
| Segment |
| PDU send |
| Streaming Media | 视频编解码算法 | MPEG |
| H264 |
| 流媒体传输协议 | 资源预留协议RSVP |
| 实时传输协议RTP |
| 实时传输控制协议RTCP |
| 实时流协议RTSP |

### 高通量应用典型程序特征总结

通过对三类高通量应用的分析，总结出了一些高通量应用最基本的程序特征。从分析结果可以看出，不同类型的高通量应用在程序特征方面表现出了较为相似的特征。

#### 数据处理类高通量应用程序特征总结

通过对数据处理类高通量应用的分析，可以得到此类高通量应用的如下特性：

**1.作业具有并发处理特性，支持高并发的系统可以提高处理能力。**数据处理类高通量应用中的大部分Workload可以通过对所处理的大数据进行切分的方式来实现并行处理，每一部分数据用一个作业来处理，这样可以多节点并发处理。此过程可以使用MapReduce等并行编程模型来辅助完成。

**2.作业之间耦合性低，作业节点之间通信量小。**数据处理类应用中的数据都是扁平化的，数据块之间逻辑上的关系较小，即数据之间耦合性低，所以，作业处理节点之间通信量较小。

**3.访存需求比较大，较小的浮点计算量。**数据处理类高通量应用多使用统计归纳和数据挖掘等方面的技术，由于需要对大量数据进行操作，而对每一数据的操作相对简单，因而访存需求相对较大。而由于很少需要进行复杂的计算，没有科学计算的需求，所以计算量比较小，几乎没有浮点计算。

**4.处理的数据多为文本数据，以字符或字符串为主，访存位宽偏小。**数据中心的数据处理类高通量应用多用于文本类数据的分析，以字符或字符串为主，与传统应用常见的32bit或64bit访存位宽相比，此类应用访存位宽偏小，8bit位宽的访存量占到不可忽视的比例。

#### 数据服务类高通量应用程序特征总结

从应用层面对数据服务类高通量应用进行分析，可以得到其具有的程序特征：

**1.作业具有天然的并发性，支持高并发的系统可以提高处理能力。**数据服务类高通量应用的一个作业就是对一个用户请求的处理和响应，可以并发处理。

**2.作业具有独立性，作业节点之间通信量小。**用户的请求都是独立的，不同的作业之间不存在业务逻辑，不同的请求之间不存在交互，所以具有良好的独立性。

**3.较高的访存计算比，较小的浮点计算量。**数据服务类高通量应用对用户请求的处理工作主要是根据不同的用户请求，查询相应的数据库或获取远程服务器资源，将结果返回给用户。其中，后台的处理过程可能会涉及到一些数据处理的工作。可以看出，此类应用对访存通路需求比较高，而很少有计算方面的需求，特别是浮点计算几乎没有。

**4.处理的数据多为文本数据，以字符或字符串为主，访存位宽偏小。**用户请求多以文本的方式进行，以字符或字符串处理为主，与传统应用常见的32bit或64bit访存位宽相比，较小位宽的访存量占到不可忽视的比例。

#### 实时交互类高通量应用程序特征总结

通过分析可以总结，实时交互类高通量具有如下程序特征：

**1.作业具有并发性。**这里的一个作业是指对一个用户维持在线状态并提供服务。这种作业具有天然的并发性。

**2.作业之间低耦合。**不同的用户提供的实时服务是相对独立的，因而作业之间耦合性低，交互通信较少。

**3.高访存计算比。**经过分析可以发现，实时交互类高通量应用对用户进行服务的重点是数据传输，基于不同的协议保证传输质量，因而，数据的访存较多而计算较少，需要系统提供较大的访存带宽或者较多的访存通路。

**4.数据量大与不规则的数据访问导致空间局部性较差，Cache失效率高。**由于需要同时服务于不同的用户，导致了访存的不规则性。

**5.访存位宽多样化，且偏小。**此类应用的用户数据以协议数据包的方式存在，而数据包中的数据多是以字节为单位来组织的，因而，与传统的应用常见的32bit或64bit访存位宽相比，此类应用访存位宽呈现多样化，且整体来说偏小，8bit和16bit位宽占到不可忽略的比例。

## 面向高通量处理器的Benchmark的实现

一套完整的Benchmark集需要对第三章所分析的所有高通量应用针对目标平台实现相应的Benchmark，而考虑到工程量的问题，本文不会对所有高通量应用进行实现，只能选取部分具有一定代表性的应用进行实现。

**本文所实现的Benchmark的选取主要基于以下原则：**

1.从三类不同的高通量应用中选取代表应用。根据分析可知，每一类高通量应用在应用特征和性能指标方面都有很大的相似性，因而，可以从每一类高通量应用中选取一个应用领域，来代表此类应用。

2.选取的应用是使用量较大的。要保证从每一大类高通量应用中选取出来的应用具有代表性，重要的一点是此应用在数据中心中的使用量较大，才能真实反映应用对平台的需求。

3.结合作者在实验室项目中已经做的工作。结合本人之前在实验室项目中已有的科研基础，可以更高效、高质量的完成相应的Benchmark。

基于以上三点，本文选取了如下三个应用作为高通量应用的Benchmark。

数据处理类选取了Big Data Analytics应用，因为Big Data Analytics中的算法都是是大数据处理中的基础算法，使用量非常广泛。而这些算法之间是相对独立的，各个算法所反映出的应用特点、程序特性和对硬件系统的需求又很相似，因而可以选取其中最为常见的几个算法来实现，而不需要实现所有算法。本文在权衡之后选取了Wordcount、Terasort、Kmeans和Grep四个算法来实现Benchmark。

数据服务类选取了Web Search应用中的响应用户请求部分来实现Benchmark，因为这一部分的工作特点最符合数据服务类高通量应用的特点。其中几个Workload构成了一个有机整体，需要同时运行才能实现一个完整的功能，因此，最终实现的Benchmark是这几个Workload构成的一个完整测试程序。

实时交互类选取了Radio Network Controller中的用户面数据处理流程部分，这一部分的工作具有典型的实时交互类高通量应用的特点，而且作为无线通信中的核心部分，使用量是相当大的，具有很大的代表性。此Benchmark主要是根据协议的定义，模拟数据包处理流程，各个Workload之间是一个有机的整体，最终构成一个测试程序作为Benchmark。

下表对选取的三个应用领域进行了总结。

表2高通量Micro-Benchmark

|  |  |  |  |
| --- | --- | --- | --- |
| 应用 | 核心Workloads | 类型 | 高通量需求指标 |
| Big Data Analytics | Wordcount | 数据处理类 | 单位时间内能够处理的数据量 |
| Grep |
| Terasort |
| K-means |
| Web Search | Query Parse | 数据服务类 | 单位时间内能够响应的请求数 |
| Database Query |
| Bool Operation |
| Term Weight |
| Document Weight |
| MSet |
| Radio Network Controller | SDU Receive | 实时交互类 | 能够支持的同时在线的用户数 |
| MACD Schedule |
| Entity Query |
| Segment |
| PDU Send |

### 提出基于线程的作业处理节点并行化模型

高通量处理器最重要的目标是尽量提高单片处理器的吞吐效率。为了达到此目的，高通量处理器需要能够支持尽量多的作业同时运行。而在处理器级别能够并行工作的载体是线程，因此，要实现用于高通量处理器的高通量应用的Benchmark，需要将不同的作业处理节点用线程来实现。

线程在高通量处理器上的调度，由对应与高通量处理器结构的Runtime来负责，而在应用层面（Benchmark），只需要保证不同作业的节点是线程即可。

因而，本文提出了一种适用于面向高通量处理器的Benchmark的编程模型：基于线程的作业处理节点并行化模型。

此模型的关键点是处理作业的节点是基于线程实现的。针对三种不同的高通量应用的应用特点，本文提出了其该有的并行化模型结构。

### 数据处理类高通量应用Benchmark实现

#### 数据处理类高通量应用多节点并行化模型

当前的数据中心数据处理类高通量应用多采用分布式处理的方式，将大量数据分块，每块数据的处理作为一个作业，交给不同的处理节点进行处理。较常用的一种框架是MapReduce框架，大部分数据处理类应用都可以用这个框架来实现。MapReduce框架中的作业就是Map和Reduce两个过程。

数据处理类高通量应用的基于线程的作业处理节点模型就是实现一个基于线程的MapReduce编程框架，一个处理节点就是一个线程，每个Map作业和Reduce作业的处理交由线程来完成。

模型结构如图2所示



图2数据处理类高通量应用并行化模型

当线程创建时，会运行一个线程体，被一个信号量阻塞，不执行具体的操作。当用户程序执行到Run\_Map 时，DPUMRLIB 内部会调用到Start\_Workers，打开信号量。这时线程就会从Task Queue 中取任务，运行的函数是通过函数指针传入的，处理的数据的位置，长度等信息也记录在Task 中。运行完一个任务后，线程会再从Queue 中取一个任务来执行。直到Task Queue 中的任务被执行完毕，线程池中的线程再次进入Wait 状态，等待下一阶段(Reduce 或Merge)的Start\_Workers。

这种设计的好处在于，即使Task 的Workload 都不相同，但每个物理线程执行的工作量是大致相同的，有利于均衡。另外，将线程处理的数据记录在Task 中，并将线程执行的任务用函数指针表示增加了灵活性，线程可以在创建之后完成多种任务直到线程被停止。

#### 完成了Big Data Analytics核心workload的实现

BigDataAnalytics应用的实现主要选取了四个最为典型的算法，每个算法是独立的，完成不同的大数据分析功能。

### 数据服务类高通量应用Benchmark实现

数据服务类高通量应用的Benchmark选取的是搜索引擎中响应用户查询请求部分的核心workload来进行实现的。

#### 数据服务类高通量应用多节点并行化模型

数据服务类应用多为C/S结构中的服务器端程序，数据服务类高通量应用的基于线程的作业处理节点模型结构如图所示：

1.一个Listen Thread用于监听请求

2.线程池，大量线程作为作业处理节点

3.每个线程对应一个Job Queue，用于缓存Listen Thread接收到的用户请求

模型结构如图3所示。

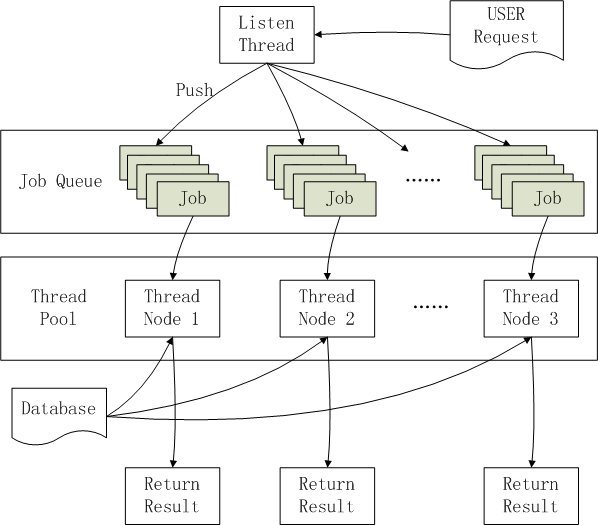
****

图3数据服务类高通量应用并行化模型

#### 完成了Web Search核心workload实现

数据服务类应用的Benchmark选取了搜索引擎系统中用户查询响应部分的应用来实现，此部分具有典型的数据服务类应用的特点和需求。

Benchmark中实现的几个Workload，包括Query Parse、Database Query、Bool Operation、Term Weight、Document Weight、MSet等，是一个完整的用户查询响应的处理流程，Benchmark代码实现时，作为一个可执程序来实现。

流程图如图 4所示。



图 4 Search各个Workload之间关系图

### 实时交互类高通量应用Benchmark实现

#### 实时交互类高通量应用多节点并行化模型

实时交互类应用每个作业的主要工作是按照协议处理用户数据。数据服务类高通量应用的基于线程的作业处理节点并行化模型结构如图所示。

1.每个用户注册之后都会有与具体协议相关的实例表，处理数据时需要查询

2.线程池，大量线程作为作业处理节点

3.每个线程节点对应多个用户，轮询处理每个用户的数据包buffer，处理时需要查询相应的实例表

模型结构如图5所示。

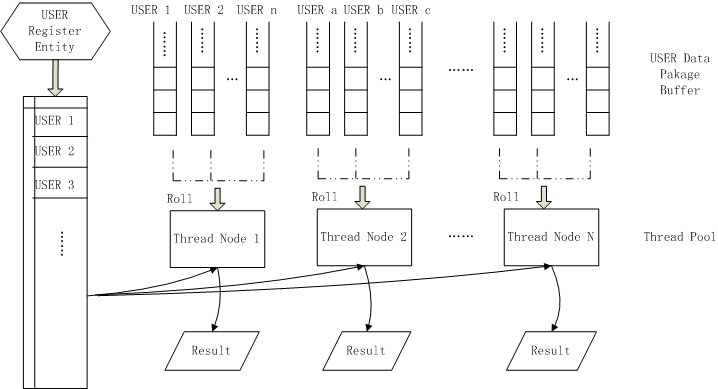
****

图5实时交互类高通量应用并行化模型

#### 完成了Radio Network Controller核心Workload的实现

无线网络控制器（RNC）是3G移动通信陆地无线接入网中数据处理的核心部分。它包含用户面和控制面。控制面主要控制初始化、资源的分配以及数据处理结束后的资源释放等；用户面负责对接入用户的数据包接收、存储、数据的协议处理和转发等，整个用户面的处理流程是按照协议来处理用户数据的。用户面是RNC中具有高通量需求的部分，本文主要对用户面的工作流程和涉及到的Workload进行Benchmark设计。

通过对RNC用户面协议进行解读和对商用代码的分析，可以提取出用户面协议处理流程中最核心的部分。主要可以分成五个Workload：SDU Receive，MACD Schedule，Entity Query，Segment和PDU Send。

最终，应用系统的吞吐能力体现在保证实时性的前提下，能够支持的同时在线的用户数。

## 实验评估

### 对Benchmark程序特性进行了实验评估

本实验部分主要用于验证所设计的Benchmark是否能正确反映出上文分析得到的高通量应用应该具有的应用特征。

#### 作业的并发性评估

高通量应用的一个重要特点是能够多节点并发处理作业，高通量处理器的一个重要目的是在线程级别，支持高通量应用的大规模并发作业，所以，处理器的吞吐效率与用于处理作业的节点数量的关系，可以体现出所设计的Benchmark是否具有良好的并发性，从而可以验证面向高通量处理器的Benchmark设计的正确性和有效性。如图 6、图 7和图 8是实验结果。

图 6 数据处理类高通量应用归一化单位时间处理数据量与线程数的关系

图 7 Search但闻时间处理请求数量与线程数的关系

图 8 RNC用户面Benchmark支持用户数与线程数之间的关系

实验数据可分析出，测试程序具有较好的并发性，提高处理机的并行能力，能够提高吞吐效率。本文所实现的Benchmark正确反映出了高通量应用应该具有的良好的并发性。

#### 作业之间耦合性评估

高通量应用的另一个显著特性是作业之间耦合性较低，这也是高通量应用与其他应用类型的一个显著区别。因此，所设计的Benchmark需要能够有效的反映出这个特性。

由于本文所实现的Benchmark是共享内存并行编程模型的，每个作业是由一个线程来处理，线程跑在多核或者众核处理器之上，因此，核间共享数据量的大小，可以反映出作业之间耦合性的大小。

图 9 高通量Benchmark各测试程序核间共享数据量占访存指令数的比例

如图直观的显示了核间共享数据量占访存指令总数的比例。从实验结果可以看出，高通量应用的Benchmark核间共享数据量占访存指令总数的比例是极小的，其中Terasort核间共享数据量所占比例最大，这是由于Terasort算法中，有较大的部分需要多线程之间共享数据，但是，核间共享数据量也只有不到0.2%，而其他几个算法所占比例几乎是可以忽略的。

所以，此Benchmark正确反映出了高通量应用各作业之间耦合性低的特点。

#### 访存需求

高通量应用主要是面向数据处理和服务，而且数据量大，因而具有较大的访存需求，而计算量较小，与传统的高性能应用相比，具有相对较大的访存计算比。所设计的Benchmark需要满足这个特性。

本文对所提取和实现的高通量Benchmark进行了访存指令的统计，如表所示。

表 3 高通量Benchmark各测试程序访存指令数所占比例

|  |  |  |
| --- | --- | --- |
|  | 线程数 | 访存指令数所占比例（%） |
| Wordcount | 2 | 56.17 |
| 4 | 73.34 |
| 8 | 89.62 |
| Terasort | 2 | 55.73 |
| 4 | 69.95 |
| 8 | 76.64 |
| Kmeans | 2 | 34.33 |
| 4 | 47.86 |
| 8 | 59.70 |
| Grep | 2 | 25.97 |
| 4 | 41.03 |
| 8 | 51.85 |
| Search | 2 | 46.62 |
| 4 | 65.63 |
| 8 | 74.99 |
| RNC | 2 | 54.34 |
| 4 | 53.03 |
| 8 | 52.46 |

从实验数据可以看出，高通量Benchmark的各个测试程序，访存需求所占比例是比较高的。

Linpack是典型的高性能计算Benchmark，在同样的硬件平台上，对Linpack进行测试访存请求比例的测试，4路并行和8路并行，访存量所占比例分别为35%和36%。所以，与典型的高性能Benchmark相比较，本文所实现的高通量Benchmark具有较高的访存需求。

#### Cache效率的评估

这个特性主要反映在数据服务类和实时交互类两种高通量应用中。这两类应用由于需要处理的是针对不同用户的数据，数据具有很强的离散性，在大块地址空间中进行非规则的访问，导致数据的空间局部性差，因而，Cache失效率较大。

使用Intel的硬件性能评估软件VTune进行实验测试，得到了Benchmark的L1 Cache Hit Rate、L2 Cache Hit Rate和LLC Cache Hit Rate三个指标，作为对比，选取Splash2作为对比，选取Splash2中的LU、Cholesky、Radix、Ocean、Radiosity和Barnes六个算法或应用进行相应指标测试。如图 10、图 11和图 12是实验结果。

图 10 L1 Cache Hit Rate实验结果

图 11 L2 Cache Hit Rate实验结果

图 12 LLC Cahce Hit Rate实验结果

从实验结果可以看到，两种类型的应用的Benchmark的L1 Cache Hit Rate指都很高，都在90以上，没有明显的差别。L1 Cache Hit Rate值在90%也属于正常范围。

而对于L2 Cahce和L3 Cache，高通量类应用的Benchmark有命中率比Splash2各个测试程序的命中率明显要小。主要体现在数据处理类和实时交互类两类应用的Benchmark。对于L2 Cache，这两类应用的命中率在60%以下，而Splash2各个测试程序的命中率都保持在70%以上。对于LLC Cache，高通量应用部分测试程序的命中率很小，在60%以下，而Splash各个测试程序LLC的Cache命中率保持在90%以上。

总体来说，与传统共享内存类应用，高通量应用Benchmark在L2和L3的Cache命中率方面表现出了较小的值。

#### 访存宽度

传统领域的高性能应用多为数值计算，因而访存宽度多以64bit或32bit为主。

与传统的应用领域不同，高通量应用由于数据格式的为多样化和不规则性，如文本数据、图数据、K-V格式数据、各类数据包等，导致访存宽度的多样化。而且，较小访存宽度的比例占到不可忽视的比例。

SPLASH2是传统并行应用领域中典型的Benchmark，是斯坦福大学推出的共享存储并行应用Benchmark，本文以这此Benchmark为对比，来比较高通量应用与传统的共享存储并行应用在访存宽度方面的差别，同时验证本文所设计的面向高通量应用的Benchmark的有效性。

图 13 访存宽度实验结果图

从图表可以看出，传统并行应用的访存宽度多集中于8B和4B，而高通量应用的访存具有多样化和不规则性。

高通量应用中1B和2B的访存占了较大的比例，这种较小的访存宽度用于传统的访存结构设计的系统中会导致带宽资源的浪费，对高通量处理器访存设计提出新的要求。

实验结论与前文分析结论相符，本文所设计的高通量应用Benchmark良好的反映出了高通量应用在访存宽度方面的特征。

### 对Tilera TILE-Gx处理和Intel Xeon处理器进行评估

TILE-Gx处理器是Tilera公司推出的一款众核处理器，TILE-Gx的目的是将上百个处理器核集中到一颗处理器上。TILE-Gx使用了一种新的Mesh网络，被称为iMesh，iMesh是一种五层网络结构，这一设计使得各个处理器核之间能够高效通信，有效的解决了众核处理器中核间通信的问题。TILE-Gx处理器达到了处理效率更高，并发处理能力更强，能耗更小和扩展性更强等方面的效果。而处理器中的每一个核处理能力有限，尤其对浮点计算的支持能力很弱。这是由于TILE-Gx更关注于网络、视频和云计算等具有大数据和高通量应用特点的应用领域，而不是针对高性能科学计算进行的设计。

本文实验所采用的是TILE-Gx系列中的TILE-Gx 8036处理器，此处理器有36个核，L1和L2两级Cache是独立的，L3是共享的。对TILE-Gx和Xeon两种处理器并行加速能力进行实验测试，实验结果如

图 14 Xeon处理器对数据处理类高通量应用Benchmark各测试程序的并行加速能力

图 15 TILE-Gx处理器对数据处理类高通量应用Benchmark各测试程序的并行加速能力

图 16 Xeon和TILE-Gx两种处理器对Search的并行加速能力比较

图 17 Xeon和TILE-Gx两种处理器对RNC的并行加速能力比较

综上实验结果可以看出，对于高通量类应用的Benchmark，Tilegx处理器比Xeon处理器具有更好的并行加速比和并行处理能力。这与Tilegx的应用定位和具体结构设计是有关的。其结构设计更适应高通量类应用中大量作业、规模较小、耦合性较低的这种特点。

这也从另一个角度佐证了本文所抽取和实现的Benchmark是合理有效的。

# 下一步工作计划和内容，预计答辩时间

论文初稿已经完成，接下来做一些修改和完善，预计答辩时间在5月中旬。

# 科研项目参与情况

**RNC用户面程序Benchmark的研究**

根据3G WCDMA协议，分析商用WCDMA代码，提取大话务量应用场景中WCDMA用户面工作的关键数据处理流程；

设计和实现模拟关键数据处理流程的WCDMA用户面Benchmark；

分析此Benchmark在通用多核处理器上运行的性能，确定限制无线通信协议性能提升的硬件瓶颈，为高通量处理器设计提供应用层依据。

**Micro-Benchmark研究**

基于开源的MapReduce编程模型项目设计多核处理器平台上的MapReduce编程库；

基于多核平台上的MapReduce编程库，实现多线程并行版Wordcount和Terasort等典型的MapReduce应用；

用Intel VTune等工具，分析通用的多核处理器的性能和微结构瓶颈，对高通量处理器的设计提供依据。

**Search Engine-Benchmark研究**

调研Hadoop和基于Hadoop的Nutch搜索引擎，调研开源搜索引擎Sphinx、Xapian等，确立设计方案；

基于Xapian设计多线程并行版Search-Benchmark，包括建立索引库和搜索响应两部分，关注高并发处理能力；

用VTune等性能分析工具，分析主流商用多核处理器的硬件瓶颈，为高通量处理器的设计提供应用层依据。

**基于异构平台的多线程代码开发与移植**

安装和调试GPU，安装和调试OpenCL库；

基于OpenCL库提供的API，移植MapReduce框架的应用程序，程序中的数据处理部分由GPU并行执行。

**ARMv8处理器核模拟软件开发**

在开源模拟器软件QEMU之上实现ARMv8指令集，即完成ARMv8指令到x86指令的二进制翻译工作；

完成ARMv8指令集译码树的实现；

完成ARMv8指令集在QEMU中的系统调用部分的设计；

针对业界通用的Benchmark完成对ARMv8处理器核模拟软件的测试和调试。

# 课程学习情况

已顺利完成了课程学习要求，取得了响应的学分。平均成绩在82分以上。

