# 题目

基于Hadoop的图像分析分布式处理研究

# 选题的科学意义和应用前景

在图像处理方面,各个应用领域图像数量增长十分迅速，从天文观测到对地遥感，从电子商务到社交网站，每天都在产生TB级的图像，此外图像质量的不断提高和需要从图像中获得越来越多的信息，使得其存储量和计算量也越来越大。图像文件相比于文本文件有占用空间大，处理速度慢的特点，因此提高图像的处理效率对于具体应用来讲，有着重要的理论意义和实用价值，而如何应对日益增多的海量数据，并进行有效的存储和快速处理,成为我们面临的主要难题。

由于大规模图像数据的存储和计算都需要计算机具有强大的计算性能。但大型机、巨型机价格极为昂贵,以我国最新研制的曙光4000A集群式超级计算机为例,其峰值运算速度预计将达到8万亿次/秒,但其价格接近亿元人民币,因此不能得到普遍应用。解决大规模计算问题不能依赖于制造越来越大型的服务器，一种替代方案已经普及，利用相对廉价的低端或商用机器通过高速局域网构建高性能的并行分布式系统[1]，与传统的超级计算机相比,具有较高的性价比和良好的可扩展性,可以满足不同规模的大型计算问题。从而被各阶层用户所接受,正逐步成为并行编程[1]的主要平台。

Hadoop集群恰恰是最盛行的分布式系统，它是在同一地点用网络互连的一组并行的通用计算机，这些机器通常是相对同构的Linux服务器，数据存储和处理都发生在这个机器“云”中。数据会被划分为较小的块通过Hadoop分布式文件系统（HDFS）分布在集群内多台机器上，使用适度的复制备份，集群可以并行读取数据，提供很高的吞吐量。Hadoop集群相对于单机系统效率更高，价格更加低廉合理。

图像并行处理技术是图像处理的一个重要方面，图像并行处理技术的应用是多方面和广泛的，因为并行确实是提高图像处理的速度的最有效技术，其发展水平一直受到图像领域的关注。但这需要付出一定的代价，包含硬件设备上和开发上的代价，图像并行处理技术的发展难度很大，这种难度不仅在于图像并行处理系统的硬件及系统结构本身，以及它对计算机技术及集成电路技术的依赖，而且在于实际应用中的复杂性和应用部门对系统价格的承受力。Hadoop很好地降低了代价。所以研究基于Hadoop的图像处理问题，尤其是实现基本的分布式图像处理，并减少图像处理的时间很有价值，实际应用系统也将产生巨大的经济效益和社会效益。

# 背景科研项目情况简介

# 学位论文主要研究内容

本学位论文主要包括以下几个方面的研究内容：

1、hadoop集群平台构建：将采用ClouderaManager对Hadoop集群进行管理。进行Hadoop平台和HIPI配置。

2、Hadoop平台技术研究：主要研究Hadoop分布式文件系统HDFS与分布式计算框架MapReduce。

3、图像处理算法的并行化理论研究：主要研究图像处理算法的并行化，结合数字图像处理理论与HIPI开放式API,设计了基于MapReduce图像处理的类型和格式，讨论了图像的分布式处理的相关技术，编写Mapper和Reducer，运行算法的MapReduce程序，实现了图像处理的并行化。

4、实验验证：最后实现了Hadoop HIPI图像处理，通过实验验证该设计的可行性。

# 预期解决的主要问题

（对每个预期解决的问题介绍其难点所在、国内外研究的现状和趋势、解决问题的基本思路和技术路线、预期解决到什么程度）

问题1：Hadoop默认处理文本文件，而图像文件的保存和处理方式与普通文本文件不一样，且Hadoop没有自带图像处理接口。需要根据MapReduce模型数据流的特征，设计了支持图像文件的Hadoop数据类型。

问题2：Hadoop框架很适合大文件的操作，我们平时使用的图片的文件都比较小，如果这种小文件不经过处理直接上传，会十分影响性能。本文采用第三方开发的图像处理接口HIPI（Hadoop Image Processing Interface），它将来源的图像进行合并处理，并存储到HDFS上再进行MapReduce处理。

问题3：基于CDH的Hadoop集群的搭建。使用ClouderaManager实现对Hadoop集群的服务管理、任务分配、信息监控等。

问题4：HDFS文件系统的配置和安装。在集群中布置HDFS文件系统，并修改相应配置文件。

问题5：图像处理、分析的相关算法的MapReduce化。要求MapReduce编程实现对图像的相应处理，如图像数据的上传、下载、合并及灰度处理。

# 开题条件

（包括学术条件、设备条件、经费概算及其落实情况）

学术条件：

课题组在HDFS文件系统、MapReduce并行处理、大数据处理等方面积累了丰富的研究经验。课题组与国内外该领域的一些专家、学者建立了很好的沟通和合作，从而便于及时跟踪该领域的最新发展动态，以促进项目的研究。

设备条件：

学校实验室具有有性能良好计算机，提供了优秀的仿真实验条件。

# 文献综述

（通过对文献的整理和归纳，对应“学位论文主要研究内容”一栏所列出的问题，**介绍国内外学者对这些问题的研究结果及对其前景的看法。**）

Sridhar Vemula等人提出了Hadoop Image Processing Framework图像处理框架，就是为了图像处理的开发人员无需掌握技术细节就能够应用Hadoop的MapReduce框架，并且避免在他们的代码中接任复杂的附加代码。HIPI抽象了Hadoop系统的技术细节，便于实现计算机视觉的很多技术。张良将等人设计了基于 MapReduce 图像处理的类型和格式，实现了图像处理的并行化。

### 1、Hadoop集群平台构建：

Hadoop是Apache软件基金的开源项目，模仿和实现了Google云计算系统的主要技术，使用Java语言编写，可移植性强。Hadoop实现的云计算平台[3]，为用户提供一种分布式计算和分布式存储的编程环境，该技术属于云体系中的一种PaaS(平台即服务)[4]。将采用Cloudera Manager对Hadoop集群进行管理。进行Hadoop平台和HIPI配置。

Hadoop有三种安装模式:本地(单机)模式、伪分布模式和全分布模式。本地模式主要是用于开发调试MapReduce程序的应用逻辑。伪分布模式在单机模式之上添加了代码的调试功能，可以运行HDFS，可以与其他的守护进程交互等功能。下面详细的介绍安装过程:

硬件基本配置要求:CPU(Intel(Ｒ)Core(TM)i5－3470CPU@3.20GHz)，内存(4G)，硬盘(1T); 安装软件要求包括Linux操作系统，远程登录协议(SSH Secure Shell)，JDK和Hadoop分布式框架等。

安装一个Hadoop集群时，需要指定一个服务器作为主节点驻留NameNode和JobTracker的守护进程。它将作为一个基站，负责联络并激活所有从节点上的DataNode和TaskTracker守护进程。因此，我们需要为主节点定制一种手段，使它能够远程地访问集群中的每个节点。Hadoop使用了无口令的SSH协议。SSH采用标准的公钥加密来生成一对用户验证密钥。公钥被存储在集群的每个节点上，私钥则由主节点在试图访问远端节点时发送过来。结合这两段信息，目标机可以对这次登录尝试进行验证。

安装SSH（Secure Shell）需要定义一个公共账号用于管理Hadoop集群。检查验证SSH安装，若没有则下载OpenSSH源码安装。安装完成后，使用主节点上的ssh-keygen生成一个RSA密钥对，避免输入口令，并将公钥分布并登录验证。

配置Hadoop时需要配置所有节点的Java位置，即在hadoop-env.sh中定义JAVA\_HOME。设置主要包含在XML配置文件中:core-site.xml，hdfs-site.xml与mapred-site.xml。

### 2、Hadoop平台技术研究：

Hadoop是项目的总称，包含3个子项目，分别是MapReduce、HDFS (Hadoop Distributed File System)和Hadoop Common。并行计算框架MapReduce是Google MapReduce[5]的开源实现，分布式文件系统HDFS是Google GFS[6]的开源实现。

#### HDFS

HDFS是管理网络中跨多台计算机存储的文件系统，是一个高度容错性和可扩展性的系统，适合部署在廉价的普通商业机器上。可以存储一个普通的文件存储系统无法存储的大文件，在物理空间上它將一个大文件分为大小固定的若干个数据块(通常为64MB)，存放在集群中每一个节点上，在逻辑空间上则是以一个文件存储。在读取数据时HDFS可以实现并行操作，所以提高了其在读取数据时的吞吐量，并且支持流失数据访问。

HDFS把Hadoop云平台中的节点分成两类：NameNode( 唯一的 ) 和 DataNode。NameNode 管理文件系统的命名空间，维护着文件系统树及整棵树内所有的文件和目录。NameNode 也记录着每个文件中各个块所在的数据节点信息，这些信息会在系统启动时由数据节点重建。DataNode 是文件系统的工作节点，它们根据需要存储并检索数据块，定期向 NameNode 发送它们所存储的块的列表。DataNode 用数据块作为最基本的存储单位，默认数据块大小为 64 MB。

#### MapReduce

是什么？

MapReduce是Google提出的一个软件框架，基于它写出来的应用程序能够运行在由上千个普通商业机器组成的集群上，并以一种可靠容错的方式并行处理大规模数据集。

怎么做？工作原理

MapReduce并行计算模式将数据处理任务抽象为两个阶段：Map(映射)和Reduce(规约)，Map主要是完成数据的过滤工作，Reduce主要完成数据的聚集工作。Hadoop将MapReduce的输入数据划分成等长的小数据块，称为输入分片。Hadoop为每个分片构建一个Map任务，并由该任务来运行用户自定义的Map函数从而处理分片中的每条记录。每个Map任务都会对其输出进行分区，进而为每个Reduce任务建立一个分区。每个Reduce任务对其分区内键值对运行用户自定义的Reduce函数进行相应处理。

用户怎么用？

用户在使用这种编程模型时，可以根据相对应的编程规则来实现Map函数和Reduce函数，然后MapReduce会自动对任务进行划分以做到并行执行［5］。

执行流程

MapReduce作业的执行可以分为11个步骤，涉及4个独立的实体，它们在MapReduce作业执行过程中的主要作用如下：

1)客户端。编写MapReduce代码，配置作业，提交作业。

2)JobTracker。初始化作业，分配作业，与TaskTracker通信，协调整个作业的执行。

3)TaskTracker。保持与JobTracker的通信，在分配的数据片段上执行Map或Reduce任务。

4)HDFS。保存作业的数据、配置信息和作业结果等。

可能存在问题

MapReduce并行计算框架MapＲeduce程序通过操作键/值对来处理数据，一般形式如下:

Map:(K1，V1)→list(K2，V2)；

Reduce:(K2，list(V2))→list(K3，V3)；

MapReduce框架并不允许任意的类作为键和值的类型，只有支持序列化的类才能够在这个框架中充当键或者值。Hadoop有自己的序列化格式Writable，实现Writable接口的类可以作为值类型，而实现WritableComparable<T>接口的类既可以作为键类型也可以作为值类型。**本课题需要使用的键类型为Text，是Hadoop的org.apache.hadoop.io包中自带的，用来存储图像文件的名；值类型为Image，需要实现Writable接口，用来存储图像文件的内容。**

#### HadoopCommon

HadoopCommon是一组分布式文件系统和通用I/O的组件与接口，提供对Hadoop子项目的基本支持。HadoopCommon在0.20及以前的版本中，包含HDFS、MapReduce和其他项目公共内容，从0.21版本开始，HDFS和MapReduce被分离为独立的子项目，其余内容为HadoopCommon。

#### 联合

两种技术的联合：

通过HDFS將输入的数据集分割后，分布到集群中的每一台机器上，然后由作业跟踪节点确定Map的任务，Map的任务结束后产生中间结果存放在本地的磁盘上，为了节省Reduce过程的时间，在洗牌(Shuffling)阶段之前对结果进行一下“本地的Reduce”即Conbiner过程。在HDFS將数据分配到每个节点后，洗牌阶段是节点间唯一的通信的时间。经过洗牌阶段后进行Reduce阶段，將最终的结果输出。

### 3、图像处理算法的并行化理论研究：

#### 并行处理

并行计算(Parallel Processing)是相对于串行计算来说的，所谓并行计算分为时间上的并行和空间上的并行；时间上的并行是指流水线技术，而空间上的并行则是指用多个处理机并发地执行计算，现在的发展已趋向于空间上的并行，它是指计算机系统中能同时执行两个或多处理机的一种计算方法，处理机可同时工作于同一程序的不同方面，以实现并行处理。

并行处理[20]是针对一个具体的问题，利用它内在的并行性，设计并行算法，将其分解为相互独立、但彼此又有一定联系的若干个子问题，再分别交给各个处理器，所有处理器按照并行算法完成初始应用问题的求解。这是一种达到高性能计算的手段，能够将计算能力从单个处理器扩展到多个处理器，但它与高性能计算还是有区别的，高性能计算是指运算速度快、性能高、处理能力强的一类计算机，它提供当前最先进的计算能力，来支撑人们理解和解决复杂的计算问题。并行计算技术是实现高性能计算的主要形式。总而言之，并行处理是一种有效的强调开发计算过程中并行事件的信息处理方式。

所谓并行事件就是指具有"并行性"的事件。并行性包含三种含义:

* + - 同时性：两个或多个事件在同一时刻发生在多个资源中；
    - 并发性：两个或多个事件在同一时间间隔内发生在多个资源中；
    - 流水线：两个或多个事件发生在可能重叠的时间段内。

并行算法[25]是指在各种并行计算机上求解问题和处理数据的算法，其本质是把多任务映射到多处理机中执行，或将现实的多维问题映射到具有特定的拓扑结构的多处理机上求解。简单来说就是一些可同时执行的进程的集合，这些进程相互作用和协调，以完成对一个给定问题的求解。

并行算法的实现依赖于计算机硬件和软件环境。由于并行算法与并行处理计算机的密切相关，因此并行算法的好坏直接关系到并行处理计算机的使用效率。不同类型的并行处理计算机或集群，在其上使用的并行算法也不尽相同，并且所要解决的具体问题也不完全一样。如果一种算法所表达出来的并行度与计算机的并行度基本一致，便能提高计算机的解题效率。另一方面，并行算法的研究对并行处理计算机的发展起着巨大的推动作用，尤其是多处理系统的发展和应用将有待于其并行算法的进一步研究。

并行处理的两种基本连接模式：

1、流水线连接，流水线式把处理单元按顺序串联在一起，即一个处理单元的输出和下一个处理单元的输入相连，类似于管道Pipe。如果一个大人物可以分解成一些复杂性大致相同的小任务，而且这些小任务可以独立完成，就可以采用这种流水线结构来完成并行处理系统。在这种流水线结构里，多种任务在流水线的各级上同时执行，整个任务的速度取决于执行时间最长的子任务的执行时间。在流水线中，任何一个处理单元出现的故障都将直接破坏整个流水线的正常工作，所以在这种流水线结构里，对每一个处理单元都有很严格的要求。在采用流水线的图像处理中，整个处理节奏是以同一速率执行的。

2、并行阵列连接，是用多个处理单元组成一个并行阵列，每个处理单元都可以独立执行任务。

总的来看，流水线连接模式是依次向流水线的每个处理单元输入信息，每个处理单元以输入信息的速率周期地输出信息。并行阵列是同时向这些并行的处理单元输入信息，一旦一个完整的任务执行完毕，所有处理单元同时产生输出。这两种连接模式，其执行速度与阵列中的处理单元数量有关。采用流水线结构的图像处理系统具有3个突出的优点：首先是告诉，一旦流水线上有作业，它就像流水装配线一样，数据速率是与时钟相一致的；其次是固有的寻址方式，这意味着无需额外铺设地址线和地址发生器；第三是无需太大的额外开销即可扩展功能。流水线结构的主要缺点是缺乏灵活性，如果你想要根据地址来进行操作而不是随顺序的数据流来进行操作，那么流水线处理器就难以胜任了。流水线是以不同食物的并行处理为特征，而并行阵列是以相同事务的并行处理为特征。本可以会使用并行阵列结构，关键是划分具体任务和输入输出数据的组织。

#### 图像并行

在图像并行处理中，有两类并行性形式：流水行并行性和数据并行性。即像先图像获取、边缘锐化、二值化再细化不同步骤并行处理的流水线；还有就是分区并行处理的情况，一幅如想均匀分为4块，每一块数据由一个处理器进行处理，这种数据并行的方法在执行邻域处理算法存在图像边缘衔接的问题。另一种数据并行的方法是采用算法和存储结构同一的方法，由图像帧并行地提供算法所需的领域数据，由处理器并行地加以处理。

本课题主要研究图像处理算法的并行化，结合数字图像处理理论与HIPI开放式API,设计了基于MapReduce图像处理的类型和格式，讨论了图像的分布式处理的相关技术，编写Mapper和Reducer，运行算法的MapReduce程序，实现了图像处理的并行化。

把一个图像文件作为一个作业分片，把整个作业分片作为一条作业记录来处理。这样每个Map任务只需调用一次Map函数来处理一个图像文件，进而实现多个图像文件的并行化处理。系统只定义了一个Reduce任务，其调用Reduce函数对每个键值对进行简单的输出操作。为了实现上述功能，首先，需要设计Map和Reduce任务的键值对的类型，用于在Map函数和Reduce函数之间传递数据。其次，文件的输入和输出格式，用于图像文件和键值对类型数据之间相互转化。最后，在Map函数中实现图像处理。

本课题的主要难点之一是Hadoop没有自带图像处理接口，所以需要着重研究图像文件的输入和输出。目前，使用MapReduce编程模型来处理图像文件的主要处理方法有两种：①将图像转换成二进制串行化数据处理。首先预处理图像文件，将其转化成二进制文件，得到图像文件的串行化数据，直接使用Hadoop内置数据类型对图像文件进行处理，如郑欣杰等利用Hadoop平台MapReduce模型研究基于MapReduce的分布式光线跟踪的设计与实现；Keith Wiley等用二进制文件处理天文学方面的图像文件。在图像处理中，有很多操作为了对某个像素进行处理，通常需要利用它的相邻像素点，将图像文件的数据串行化，并不能支持这样的操作；②实现自定义的图像文件接口。如Chris Sweeney等自定义图像文件接口实现批量图像文件的信息提取，他们将图片文件看作一个数组矩阵，这样只能处理有限的图像文件类型。李倩、施霞萍根据MapReduce模型数据流的特征，设计了可以处理图片文件的Hadoop数据类型，直接针对图片文件进行处理，不仅可以进行边缘检测等要利用到相邻像素点的图像处理，还可以处理各种类型的图像文件。

对于集群计算而言，一个很重要的原则就是设法加大计算时间相对于通信时间的比重，减少通信次数甚至以计算换通信，这是因为对集群系统而言，一次通信的开销要远远大于一次计算的开销，因此要尽可能降低通信次数。基于这个原因，集群计算的并行粒度划分不能太小，因为这会增大通信的开销，如果能实现计算和通信的重叠，将会极大地提高整个程序的执行效率。Hadoop强调把代码向数据迁移，集群内部既包含数据又包含计算环境，客户端仅需要发送待执行的MapReduce程序，而程序一般都很小。数据被拆分后在集群中分布，并且尽可能让一段数据的计算发生在同一台机器上，即这段数据驻留的地方。

并行算法的设计总体上遵循PCAM(Program Complexity Analysis Methodology)的方法，它的建立分为四个步骤：

(1)任务划分：任务划分分为域分解和功能分解。域分解是分解与问题相关的数据；功能分解是按功能来进行任务分解。本文的并行程序设计采用的是域分解法，算法描述如下：一幅图像可看作是一个二维数组，图像的像素值是数组中的元素，元素个数为图像面积，如256×256大小的图像像素有256×256个。把所给任务按要求划分成P块，然后分别分配给P个节点计算机，让这P个节点同时并行运行，以上是任务能被P整除的情况，不能整除时，要把多余的部分留给管理节点处理，管理节点处理的数据范围是P个节点处理的数据范围。它的实现可以用任何一种高级语言完成，本文是在前人工作的基础上设计了基于网络工作站NOW(Net of Workstation)的集群计算机系统，它本身是一个很好的并行编程环境；

(2)通信分析：由划分产生的各个并行任务，一般情况不能独立执行，一个任务中的计算可能需要用到另外一个任务中的数据，从而产生通信要求，所谓通信就是为了进行并行计算任务之间的数据传输。在本程序的设计中，由管理节点向计算节点分发任务，广播公共信息时的通信是通信量最大的，我们尽可能地把公共信息进行封装打包一次性广播，任务分块时尽可能使任务分块明确，减少任务间的数据联系，从而减少任务运行时由于确认任务边界问题而产生的节点间的数据通信；

(3)组合：重新考察该任务划分时所做的选择，力图得到一个在某一类并行机上能有效执行的并行算法。组合的目的是通过合并小尺寸的任务来减少并行任务数，以提高效率和减少通信成本。通过实验得到能最快完成并行任务的途径，在实验中发现，当集群节点数接近某一值时并行效率达到最高，节点太少时任务分块变大，单个处理时间变长；节点太多时，由于集群采用的是千兆以太网，通信量大时可能产生通信瓶颈，造成通信阻塞，降低并行效率；

(4)映射：映射阶段是将在组合阶段合并的任务分配给各个处理机，其主要目的是减少算法的总执行时间。

#### 性能评价

(1) 算法并行度[27]：并行计算机执行一个程序可以在执行过程的不同时间范围内使用不同数目的处理机，通常将每个时间范围内用来执行程序的处理机数目称为并行度(Degree of Parallelism,DOP)。并行度反映了软件并行性与硬件并行性匹配的程度，并行度在一定程度上反映了一个并行算法究竟能在多大程度上并行执行。

(2) 算法复杂度：算法复杂度是指其所含的工作量，例如串行算法的复杂度包括时间复杂度和空间复杂度。算法复杂度是求解问题规模的函数[28]。一般而言，处理 N ×N的图像所需的计算复杂度为 。如果涉及到等级更高要求的计算时计算复杂性还要提高。

(3) 时间复杂度：如果一个问题的大小是 n，解决这一问题的某一算法所需时间为，它是n的函数，则称 T (n)是这一算法的时间复杂度。当使用并行算法处理数字图像时，它的阶次降低，由时间复杂度看出，计算量的降低为平方关系。即：如果 N 减少为原数据的一半，则计算量降低到 1/4，可见若有并行计算被应用，图像处理的速度将会大大加快。

(4) 通讯复杂度：通讯复杂度的计算是与并行算法的设计及实现所依赖的并行体系结构相关联的，它是指某个并行算法在某种结构的并行系统上实现后，为了完成算法要求的计算需要在节点之间进行通信所耗费的时间的复杂度。它一般是问题规模n和并行系统节点个数P的函数。

(5) 运行时间：并行算法的运行时间是指算法在并行计算机上解决一个问题所需要的时间，即表示算法开始执行到结束的这一段时间，如果不同处理机不能同时开始或同时结束它们的计算，那么此算法运行的时间定义为：从第一台处理机开始执行的时间算起到最后一台处理机完成所经历的时间。

(6) 并行算法的成本：设并行系统处理机台数为P(n)，并行算法在其上的运行时间为T (n)，则并行算法的成本的定义为：C ( n)= T(n)×P(n)，这样，我们就可以根据预定成本来确定是使用更多的处理机，还是忍受较长的运行时间。

# 学位论文工作进度安排

查阅主要文献资料目录清单（40）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 作者 | 题目 | 刊物名称 | 期（卷）号 | 年份 | 起止页码 |
| 林鹏 | 基于集群系统的并行图像处理算法的设计与实现 | 中国海洋大学 |  | 2006 |  |
| 苏光大 | .图像并行处理技术 | 清华大学出版社 |  | 2002 |  |
| CockshottP,RenfrewK | ParallelImageProcessing | Springer |  | 2001 | 1194-1199 |
| 王彦明,奉国和,薛云 | 近年来Hadoop国外研究综述 | 计算机系统应用 |  | 2013 | 22(6):1-5 |
| VemulaS,CrickC | HadoopImageProcessingFramework |  |  | 2015 | 506-513 |
| [张岩](http://epub.cnki.net/kns/popup/knetsearchNew.aspx?sdb=CJFQ&sfield=%e4%bd%9c%e8%80%85&skey=%e5%bc%a0%e5%b2%a9&scode=08687667%3b25770940%3b22513481%3b);[郭松](http://epub.cnki.net/kns/popup/knetsearchNew.aspx?sdb=CJFQ&sfield=%e4%bd%9c%e8%80%85&skey=%e9%83%ad%e6%9d%be&scode=08687667%3b25770940%3b22513481%3b);[赵国海](http://epub.cnki.net/kns/popup/knetsearchNew.aspx?sdb=CJFQ&sfield=%e4%bd%9c%e8%80%85&skey=%e8%b5%b5%e5%9b%bd%e6%b5%b7&scode=08687667%3b25770940%3b22513481%3b) | 基于Hadoop的云计算试验平台搭建研究 | [沈阳师范大学学报(自然科学版)](http://epub.cnki.net/kns/Navi/ScdbBridge.aspx?DBCode=CJFD&BaseID=SYSX&UnitCode=&NaviLink=%e6%b2%88%e9%98%b3%e5%b8%88%e8%8c%83%e5%a4%a7%e5%ad%a6%e5%ad%a6%e6%8a%a5(%e8%87%aa%e7%84%b6%e7%a7%91%e5%ad%a6%e7%89%88)) |  | 2013 |  |
| MapReduce | Simplified Data Processing on Large Clusters |  |  |  |  |

