doi:10.16652/j.issn.1004-373x.2017.09.031

基于Hadoop的网络日志挖掘方案的设计

许抗震，吴　云
（贵州大学 计算机科学与技术学院，贵州 贵阳　550025）

摘　要：提出一种挖掘指数级别网络日志数据的解决思路，设计了一个高可靠的网络日志数据挖掘方案。针对现有的公开网络日志数据集，在数据预处理阶段实现了基于MapReduce的过滤算法，并且挖掘出支持企业决策的服务信息。对该方案搭建的平台进行优化操作，性能提升了3.26%，最后对方案的高可靠性、日志文件个数对平台I/O速度的影响、平台和单机在查询性能上的对比等方面做了实验。结果表明：该设计方案不仅可靠，而且随着日志文件个数的翻倍增加，读操作耗时平均增加52.58%，写操作耗时平均增加79.69%。随着日志量的增加，单机的查询耗时急剧增长，而平台的查询耗时趋于稳定。随着机器节点的增加，运算耗时以平均8.87%的速度减少。

关键词：网络日志；数据挖掘；数据清洗；Hadoop；MySQL
中图分类号：TN711-34；TP391.9　　**文献标识码：**A　　**文章编号：**1004-373X(2017)09-0115-06

Design of Web log mining scheme based on Hadoop

XU Kangzhen，WU Yun
（College of Computer Science and Technology，Guizhou University，Guiyang 550025，China）

Abstract：A thought of mining the Web log data with exponent level is put forward. A high reliability Web log data mining scheme was designed. Aiming at the available public Web log dataset，the filtering algorithm based on MapReduce was implemented in the data preprocessing stage to mine the service information supporting the enterprise decision. The platform established with this scheme is optimized，and its performance is increased by 3.26%. The effect of the scheme′s high reliability and log file quantity on the I/O speed of the platform，and the comparison of the platform with the single machine in the aspect of query performance were tested. The results show that the designed scheme is reliable，double increased with the increase of the log file quantity，the time cost of the read operation is increased by 52.58% averagely，and the time cost of the write operation is increased by 79.69%. With the increase of the log quantity，the query time cost of the single machine is increased rapidly，and the query time cost of the platform is stable. With the increase of the machine nodes，the computational time cost is decreased by 8.87% averagely.

Keywords：Web log；data mining；data filtering；Hadoop；MySQL

0　引　言

随着信息爆炸时代的到来，在日常生活中每天都会产生指数级的数据，特别是网络日志，这就必然带来一系列的问题。一方面数据存储量过于庞大而且存储资源有限，另一方面传统的计算方式使得计算过程周期过长，计算资源得不到合理分配。随着Apache公司Hadoop的诞生，利用低价的集群实现了强大的计算和海量存储，并且在网络日志挖掘这一领域出现了一些应用研究[1-2]。本文的一个目的就是研究Hadoop框架及其生态系统成员，搭建出一种基于Hadoop生态系统成员Zookeeper的高可靠(High Available，HA)集群并进行优化。

网络日志数据是一种暗数据，所谓暗数据就是大数据的一个子集，是指被用来收集、处理和存储但不做其他用途的数据，因其数据量庞大，受到技术和工具的限制，一般并不用来做分析[3]。但是网络日志数据对于企业有巨大的潜藏价值，它更能反映出用户的浏览兴趣与意图。利用数据挖掘技术挖掘出有价值的信息可以提高企业的产品质量和服务。另外，面对这种指数级别的日志数据，不但要关注数据的量的庞大，还要要求数据的质的可靠无误。网络日志数据往往很驳杂，数据中存在着错误和不一致性，需要剔除或者加以改正[4-5]。所以，

收稿日期：2016-05-13
基金项目：国家自然科学基金项目(NSF61370161)；贵州省科学技术基金项目(黔科合J字[2010]2100)；贵州大学博士基金项目(贵大人基合字(2009)029)

本文利用Hadoop平台设计并实现了基于MapReduce的分布式清洗算法，能够提取出高质量的数据集，方便进行数据挖掘。

结合Hadoop和编程思想进行自底向上分架构的设计[6]：数据存储层、业务逻辑层、应用层。最后用实验测试平台的性能，并且用某网站的日志数据做实验挖掘出有价值的信息，供企业进行决策和改善用户体验。

1 Hadoop生态系统主要成员

自从GFS[7]和MapReduce问世之后，开源项目Hadoop就逐渐形成了自己的生态系统，特别是在Hadoop 2.0出现之后，为解决企业数据快速分析和挖掘提供了强大的分布式数据平台。具体介绍如下：

HDFS：一个主从结构的分布式文件系统类似于GFS用于存储大量数据[8]。

MapReduce：Hadoop的分布式批处理框架[9]。

YARN：一种新的资源协调管理系统，可以部署上层应用，大大提高了集群的利用率[9]。

Flume：一个高可靠的分布式海量日志收集系统。

ZooKeeper：一个能够提供协调服务的分布式协调服务框架。

Sqoop：Hadoop与外界的一个数据迁移工具。

Hive：可以进行多维分析的数据仓库系统。

2 基于Hadoop的网络日志挖掘方案的设计

当前，很多网站每天都会产生指数级别的日志数据，由于这些数据是一种暗数据，很多企业并不重视，这就造成了许多具有价值的信息被埋没。而且目前的数据挖掘很多都是基于传统的关系型数据库，挖掘时间很长、效率底下，影响了业务决策。本文利用廉价的计算机集群设计搭建一种基于网络日志数据的挖掘方案来解决这些问题。

2.1 设计架构

网络日志数据具有异构性、多样性、复杂性等特点，结合网络日志数据的这些特点和编程思想把平台的设计分为三层架构，如图1所示。

数据存储层：包括日志收集系统和底层数据的存储计算，日志收集系统主要负责网站日志的收集，然后调用过滤算法进行过滤，以文件形式传入到Data Node底层数据节点上进行存储，所有的计算结果也是在数据存储层，包括清洗过的中间数据、数据挖掘的结果、日志的元数据等。

业务逻辑层：是平台的核心，工具HBase，Hive，Zookeeper不仅实现了业务逻辑的封装还简化了用户的使用，所以使用这些工具进行数据处理和数据挖掘。根据用户上网行为模型、网站流量模型、网站VIP用户模型挖掘出浏览量PV，IP数等有价值的信息。分析结果以文件形式存储在HDFS上，Hive的元数据通过用户接口CLI（command line interface）以MySQL作为存储引擎。最后使用Sqoop工具把分析结果导出到集群外的关系型数据库MySQL中。

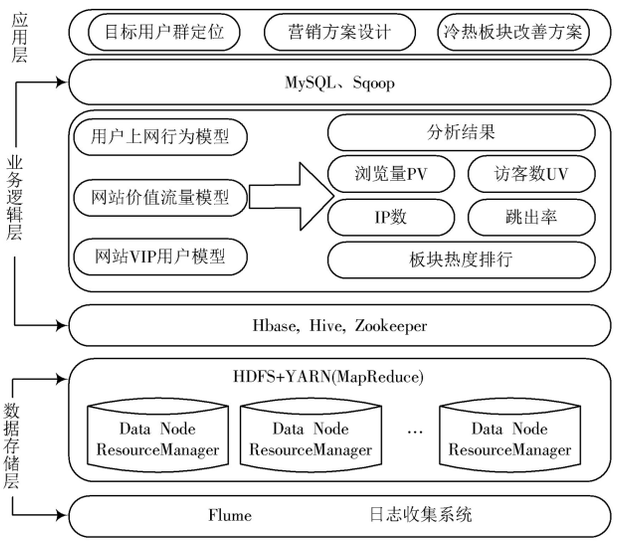

图1 方案的设计架构

应用层：利用业务逻辑层中的分析结果，例如VIP用户、板块热度、浏览量PV等用户特征和流量统计结果可以定位目标用户群，进行精准的营销方案设计，还可以针对冷清网页板块进行改善，提高网站的用户体验等决策服务支持。

2.2 节点的规划

平台在异构环境下搭建，整个集群由6台华硕的廉价机器组成，配置均为：四核3.2 GHz的i5 CPU，4 GB内存，2 TB硬盘，机器搭载的是CentOS 2.0系统。搭建的软件是Hadoop 2.0，集群中准备配置两个主节点，分别处于Active状态和Standby状态。Active状态的主节点可以对外提供服务，Standby状态的主节点并不提供服务，只是通过JournalNode进程同步Active状态节点的元数据。当其中的Active状态节点宕机，会快速自动的进行ZKFC（DFSZKFailoverController）故障转移。这样配置就保证了集群的高可靠性。平台的主机名，IP，安装的软件和进程规划如图2所示。

2.3 搭建主要步骤

（1）首先安装配置Zookeeper，然后将配置好的zookeeper拷贝到itcast05和itcast06节点下。

（2）配置HDFS时先将Hadoop添加到环境变量中，再分别修改HDFS的配置文件。

（3）修改slaves并配置免密码登录，slaves是指定子节点的位置，所以首先要配置itcast01到itcast02，itcast03，itcast04，itcast05，itcast06的免密码登录，而后配置itcast03到itcast04，itcast05，itcast06的免密码登录。

（4）安装配置HBase，Sqoop和Hive，这三个工具按照功能作用分别组成平台的数据库，数据转移模块和多维分析模块。首先需要把Sqoop安装在itcast01节点上，添加到环境变量，将数据库连接驱动拷贝到$SQOOP_HOME/lib里，配置MySQL的远程连接。然后解压下载好的hbase-0.94.2-security.tar.gz，修改相关文件保存并退出。最后解压hive-0.9.0.tar.gz，修改保存相关文件并退出。

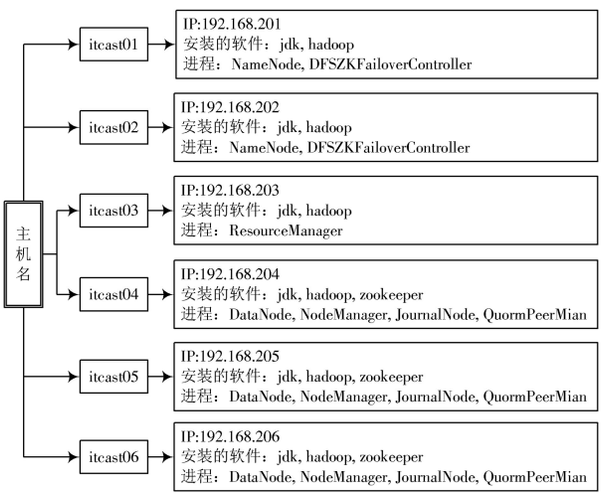

图2　节点规划

3　网络日志数据的分析与挖掘

首先利用Flume模拟日志的生成和抓取，这样和真实的生产环境一样，保证了数据搜集的真实性。然后应用设计好的基于MapReduce的清洗算法对数据进行清洗过滤[10]，设计KPI（Key Performance Indicator），使用Hive进行多维分析。最后使用Sqoop工具把结果导入MySQL中进行可视化显示。系统管理员进行集群的管理和监控，用户通过Web UI挖掘需要的信息。

3.1　数据清洗算法和KPI的设计

3.1.1　清洗算法设计

实验所用的日志数据是某网站的公开数据集合，原始数据的格式如表1所示。一共54万行数据，日志数据每行由6部分组成：用户IP、请求日期、请求方式、访问资源、访问状态、访问流量。首先把数据上传到HDFS，然后根据数据格式设计过滤算法。

从表1中可以看出，网站的日志数据格式比较杂乱，需要对数据进行清洗过滤，首先HTTP页面请求方式有很多，其中只有GET这种方式才能代表用户的行为意图，所以需要剔除其他方式。其次，返回的状态码只有2开头的才表示用户访问成功，其他噪声状态码也需要剔除。具体算法用Java实现，算法流程如图3所示。

表1　原始数据格式

日志数据格式	含义
110.52.250.143	用户IP
[30/May/2014:17:38:20…]	请求日期
GET，POST…	请求方式
/static/image…	访问资源
100~599	访问状态
1127	访问流量

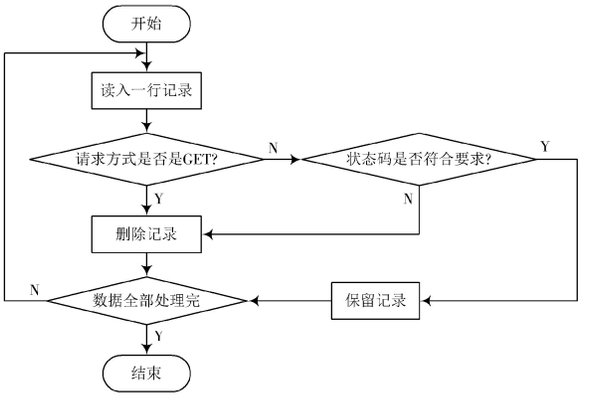

图3　过滤算法流程

在写MapReduce清洗算法之前，首先写一个名为LogFilter的类，它的作用是用来解析网络日志的行记录，这个类中包含一个方法parse，接收网络日志的一行数据，按照数据的格式拆分为用户IP、请求日期，请求方式和访问资源放一起，访问状态，访问流量。把这五部分装入一个数组返回。

```
Mapper函数{开始循环Map方法
  Map(LongWritable key, Text value) {
  value转化为字符串赋值给line;              //读入一行记录
  LogFilter类实例化调用parse返回一个数组logs;
  //logs[0]装用户IP,logs[1]请求日期等5部分数据
  if(logs[3]以"GET/static"或"GET/uc_server"开头){
    返回;}
  if(logs[3]以"POST"开头){
    跳出函数进行下一次循环;}
  if(logs[4]不是在200~299){
    跳出函数进行下一次循环;}
  序列化(logs数组)放入vl对象;          //经过过滤后的数据
  输出(key,vl);                        //作为Reduce函数的输入
  }
}
```

因为不需要排序和组合，所以Reduce函数按输入输出。

3.1.2 KPI的设计

数据挖掘和数据分析的最大不同就是，数据挖掘的目标信息并不明确，应用到的技术和算法也不确定，又因为网络日志数据中往往含有大量衡量一个网站关键的指标，所以本文根据这些关键指标来指定KPI的设计。

（1）页面浏览量PV（Page View）

定义：页面被访客浏览的次数，每个独立用户只要浏览一个页面次数就加一次[11]。

分析：网站是由一个个页面组成，所有页面浏览量的总和就是网站浏览量，反映了客户对网站的兴趣程度[11]。

公式：记录次数。

（2）IP个数

定义：一段时间内访问网站的不同IP 数量和[11]。

分析：一般来讲，IP的数量与网站的访问用户成正比。

公式：对不同IP进行计数。

（3）跳出率

定义：访问网站的某个页面后便离开的访客数占总的访客数的百分比[11]。

分析：跳出率可以衡量网站对于用户的吸引程度并且呈反比。

公式：统计一天内只出现一条记录的IP，称为跳出数 T。

跳出率= $T/PV \times 100\%$

（4）版块热度排行榜

定义：版块的访问情况排行。

分析：热度反映了用户是否喜欢网页中的某板块，可以帮助企业巩固热点版块，加强冷清版块建设。

公式：按访问次数、停留时间统计排序。

3.2 使用Hive进行数据挖掘

（1）初始化

在Hive下创建一个外部表，用来连接HDFS上的数据，并把这张外表源数据存储在MySQL中。

（2）写shell脚本

每天把HDFS的日志数据处理一次，用MapRuduce过滤数据，写入到shell中的daily.sh。

（3）分析流量排名前三的用户

在Hive中输入查询命令，并且导入一张名为PV的表中，如图4所示。

图4便是某一天流量排前三的用户，说明这几个用户不仅仅是经常浏览网站的老客户，而且也是流量贡献较多的客户，可以针对这些老客户进行个性化营销推荐。

```
hive> select * from hmbs limit 3;
OK
110.52.250.126  20130530173820  data/cache/style_1_widthauto.css?y7a    20150825
110.52.250.126  20130530173820  source/plugin/wsh_wx/img/wsh_zk.css     20150825
110.52.250.126  20130530173820  data/cache/style_1_forum_index.css?y7a  20150825
Time taken: 4.966 seconds, Fetched: 3 row(s)
hive> select * from hmbbs limit 3;
```

图4 排名前三的用户

（4）查询点击量前十的用户

在Hive中输入查询点击量命令，结果如表2所示。

表2 点击量排行前十用户

用户IP	点击量
61.50.141.7	4 855
222.133.189.179	3 942
60.10.5.65	1 889
220.181.89.156	1 877
123.147.245.79	1 571
61.135.249.210	1 378
49.72.74.77	1 160
180.173.113.181	969
122.70.237.247	805
125.45.155.27	735

可以把表2中这些用户当作VIP用户群，对这些用户群进行重点关注。

（5）查询页面浏览量排行

用Sqoop导入到MySQL中，如图5所示。

图5 网页浏览量排行

从排名可以看出哪些页面是非常受用户欢迎的，哪些是比较冷清的板块，提供给企业进行参考。

4 平台优化与实验测试

4.1 平台优化

Hadoop平台的优化主要有以下几个方面：

（1）合适的硬件选择

Namenode应选择大内存、多路多核、高主频的CPU，Datanode内存根据CPU的虚拟核数（vcore）进行公式配比。

vcore数=CPU个数×CPU核数×HT（超线程数）

内存大小=vcore×2 GB

（2）操作系统和JVM优化

首先不使用可能导致操作超时的swap分区，其次根据vm.overcommit_memory的值决定分配策略，设为2。vm.overcommit_ratio的值设定为超过物理内存的比例10%，最后关闭THP(Transparent Pages)并开启JVM重用。

（3）Hadoop参数优化

Hadoop有80多个参数和性能有关，主要是HDFS，YARN和Mapreduce优化。块大小(dfs.block.size)就是默认的64M，中间结果设置分布在多机器(mapred.local.dir)，分片大小和块大小一致。优化后的操作性能耗时减少了3.26%，如图6所示。

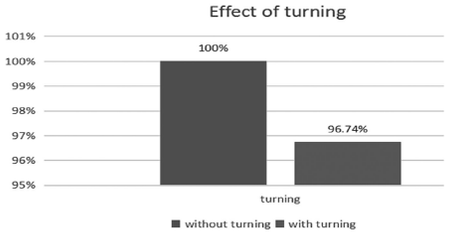

图6　优化后的性能对比

4.2　试验测试

为了减少偶然性的干扰且相对准确地测出结果，测试的数据都是四次以上取平均值。

测试一：验证平台的高可靠性

首先向hdfs上传一个文件，然后再kill掉Active的NameNode。用浏览器访问：http://192.168.1.202:50070，Standby状态的主节点就变成了Active状态，再执行命令Hadoop fs -ls/，刚才上传的文件依然存在，手动启动那个宕机的NameNode，显示NameNode 'itcast02:9000'(standby)，这就保证了平台的高可靠性。

测试二：日志文件个数对平台I/O速度的影响

首先进行总大小都为100 MB，日志文件个数分别为10，30，50的读操作，然后再进行总大小都为100 MB，日志文件个数分别为10，30，50的写操作，结果如图7所示。

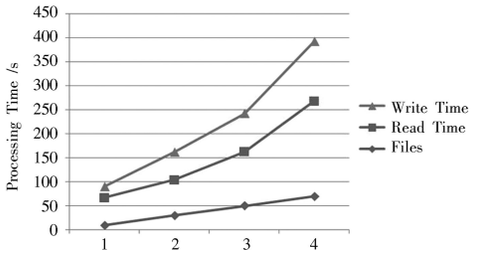

图7　I/O测试

随着文件个数的增多，读写时间明显变长，所以本平台比较适合大规模的日志文件，对于小文件处理将会花费较多时间。在文件总大小一样的情况下，随着文件个数的增加，读操作耗时平均增加52.58%，写操作耗时平均增加79.69%。

测试三：平台和单机在查询性能上的对比

测试采用一台和方案中配置完全相同的单机，分别在4万、8万、12万、16万条日志数据中查询排名前十的页面浏览量，结果如图8所示。

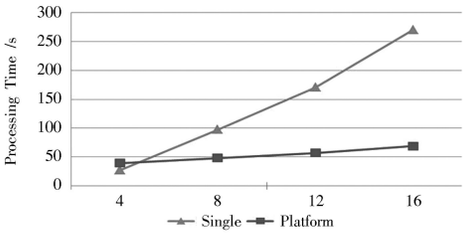

图8　单机与平台查询性能对比

从图8可以看出，当数据量很少时，用单机进行数据挖掘效率很高，但是随着数据量的增大，单机效率明显下降，而且很可能会存在内存溢出的问题，数据需要预先拆分才能完成挖掘任务，与单机相比，本文搭建的数据平台具有极高的性能查询能力，能够很好的胜任海量的网络日志挖掘。

测试四：平台的扩展性

本文搭建的平台部署了6个节点，测试时每次增加一个计算节点，组成节点个数分别为6，7，8，9的集群，查询16万条日志数据排名前十的页面浏览量所耗时间如表3所示。

表3　不同节点的耗时

节点个数	耗时 /s
6	58.254
7	50.326
8	45.275
9	43.924

由表3可以看出随着节点的增加，计算耗时以平均8.87%的速度减少，平台的高扩展性随之展现，当遇到海量数据处理时，只需要增加计算节点个数，这样就可以不用把数据切分，也避免了内存溢出等问题。同理，当数据量庞大时，也可以通过增加存储节点来解决存储问题。所以本文搭建的平台不仅拥有良好的计算扩展性，也有良好的存储扩展性，不需要进行很多复杂的配置就能解决海量日志数据处理的问题。

5　结　语

本文以Hadoop生态系统作为数据挖掘工具，搭建了一套高可靠、高扩展的数据挖掘集群，结合MySQL数

据库，在现有的公开网络日志数据集上设计了一种网络日志挖掘方案。利用MapReduce框架对原始数据集进行过滤算法的设计，清洗出高质量的目标数据，使用工具Sqoop和Hive分析暗数据，挖掘出浏览量PV排名、IP数（访客数）、流量排名前十的IP等很多有隐含价值的信息，提供给企业进行决策。实验表明，本文提出的方案有效可行。在接下来的研究中，将重点挖掘分析每日新增的访客数，浏览板块热度等信息，并且需要设计如何周期性地把明细日志导入HBase中存储，如何周期性地使用Hive进行日志的多维分析等问题。

注：本文通讯作者为吴云。

参 考 文 献

[1] YU H Y, WANG D S H. Mass log data processing and mining based on Hadoop and cloud computing [C]// Proceedings of 2012 the 7th International Conference on Computer Science and Education. Melbourne: IEEE, 2012: 197-202.

[2] MASSONET P, NAQVI S, PONSARD C, et al. A monitoring and audit logging architecture for data location compliance in federated cloud infrastructures [C]// Proceedings of 2011 IEEE International Symposium on Parallel and Distributed Processing Workshops. [S.l.]: IEEE, 2011: 1510-1517.

[3] Gartner IT Glossary. Dark data [EB/OL]. [2015-03-16]. http://www.gartner.com/it-glossary/dark-data.

[4] 郭逸重.Hadoop分布式数据清洗方案[D].广州：华南理工大学，2012.

[5] 杨富华.网络日志预处理中优化的会话识别算法[J].计算机仿真，2011(4)：123-125.

[6] 任凯，邓武，俞琰.基于大数据技术的网络日志分析系统研究[J].现代电子技术，2016，39(2)：39-41.

[7] GHEMAWAT S, GOBIOFF H, LEUNG S T. The Google file system [C]// Proceedings of 2003 the 19th ACM Symposium on Operating Systems Principles. New York: ACM, 2003: 29-43.

[8] 陈吉荣，乐嘉锦.基于Hadoop生态系统的大数据解决方案综述[J].计算机工程与科学，2013，35(10)：25-35.

[9] 顾荣，王芳芳，袁春风，等.YARM：基于MapReduce的高效可扩展的语义推理引擎[J].计算机学报，2015(1)：74-85.

[10] 杨东华，李宁宁，王宏志，等.基于任务合并的并行大数据清洗过程优化[J].计算机学报，2016(1)：97-108.

[11] 邹培忠.网站评价指标与建设现状分析[J].计算机光盘软件与应用，2012(20)：151-155.

作者简介：许抗震(1990—)，男，河南开封人，硕士研究生。主要研究领域为大数据及其应用。

吴 云(1973—)，男，博士，副教授，硕士生导师，贵州大学计算机仿真研究所所长。主要研究领域为云计算、大数据及其应用。

(上接第114页)

从图2得知，通过对少年健康体育行为数据预测，数据特征分布的收敛性较好，扰动误差较低。

4 结 语

本文采用大数据分析方法进行少年健康体育行为统计分析，指导少年体育训练管理，提出数据挖掘技术的少年健康体育行为建模分析方法。仿真结果表明，采用该方法进行少年健康体育行为应用分析，对体育关联数据挖掘的准确度较高，能有效指导少年体育管理和训练。

参 考 文 献

[1] 曾志，王晋，杜震洪，等.一种云格环境下可计算资源与服务高效调配机制[J].浙江大学学报(理学版)，2014，41(3)：353-357.

[2] 徐晓斌，张光卫，孙其博，等.一种误差可控传输均衡的WSN数据融合算法[J].电子学报，2014，42(6)：1205-1209.

[3] 罗亮，吴文峻，张飞.面向云计算数据中心的能耗建模方法[J].软件学报，2014，25(7)：1371-1387.

[4] 赵志刚，张振文，张福刚.自适应扩展的简化粒子群优化算法[J].计算机工程与应用，2011，47(18)：45-47.

[5] 刘杨，田学峰，詹志辉.粒子群优化算法惯量权重控制方法的研究[J].南京大学学报(自然科学版)，2011，47(4)：364-371.

[6] 方芳，程效军.海量散乱点云快速压缩算法[J].武汉大学学报(信息科学版)，2013，38(11)：1353-1357.

[7] MERNIK M, LIU S H, KARABOGA M D, et al. On clarifying misconceptions when comparing variants of the artificial bee colony algorithm by offering a new implementation [J]. Information sciences, 2015, 291(10): 115-127.

[8] HSIEH T J. A bacterial gene recombination algorithm for solving constrained optimization problems [J]. Applied mathematics and computation, 2014, 231(15): 187-204.

[9] KHALILI A, SAMI A. SysDetect: a systematic approach to critical state determination for industrial intrusion detection systems using Apriori algorithm [J]. Journal of process control, 2015, 32(11): 154-160.

作者简介：董世彪(1979—)，男，河南汝州人，研究生，副教授。主要从事民族传统体育教学与训练。