

# 基于海量存储云调度机制的云网络数据存储算法

钱伟强

(陕西交通职业技术学院 西安 710018)

**摘要:**针对当前云网络超宽带存储难以实现流量的分流,且存储过程中存在严重的存储效率低下及存储时延过高等问题,提出了一种基于海量存储云调度机制的云网络数据超带宽存储算法。采用周期调度及梯度优化方式,且综合考虑数据存储中使用强度,最小传输粒度等数字特征,对数据存储过程中的指纹梯度进行优化,且将该梯度引入到数据传输过程中,成功地实现了数据的流量分离,提高了数据存储效率。仿真实验表明,与当前广泛使用的超线性存储调度算法(super linear memory scheduling algorithm, SLMS)相比,本文算法的存储效率更高,能够在流量分离的前提下显著降低存储时延。

**关键词:**云网络;超带宽存储;周期调度;指纹梯度;数据粒度;流量分离

**中图分类号:** TP393 TN91 **文献标识码:** A **国家标准学科分类代码:** 520.3040

## Research on cloud data storage algorithm based on mass storage cloud scheduling

Qian Weiqiang

(Shaanxi Communication Vocational and Technical College, Xi'an 710018, China)

**Abstract:** In order to solve the current cloud storage network to realize ultra wideband traffic diversion, storage efficiency and storage delay serious problems and the storage process, we proposed a mass data storage cloud scheduling mechanism based on super cloud network bandwidth storage algorithm. The periodic scheduling and gradient optimization methods, and considering the data stored in the use of strength, the minimum size of digital transmission characteristics, fingerprint gradient data storage process was optimized, and the gradient is introduced into data transmission process, the successful implementation of the data flow separation, improve the efficiency of data storage. The simulation results show that compared with the widely used linear storage scheduling algorithm, the proposed algorithm has high storage efficiency and can greatly reduce the storage delay under the premise of traffic separation.

**Keywords:** cloud network; ultra wide bandwidth storage; periodic scheduling; fingerprint gradient; data granularity; traffic separation

### 1 引言

随着云网络技术的不断推广,网络中以流状态存在的海量数据也呈现日益增加的存储需要。由于硬件存储技术的改进,当前单网络节点的数据存储量能够达到P级别,使得数据资源日益成为国民经济增长过程中所以依赖的决定性资源之一<sup>[1]</sup>。考虑到数据存储过程中需要进行必要的资源备份及传输,因此人们对数据传输过程中的精确度及带宽需求也随之提高,通过一定方式对数据进行超宽带存储传输,成为当前研究领域中的一项目具有领先意义的技术<sup>[2]</sup>。

为提高云网络中数据存储质量,降低存储过程中存在

的技术风险,人们提出了一些实用价值较高的存储算法。如沈晴霓等人<sup>[3]</sup>提出了一种基于传输链路自适应机制的云网络数据存储算法,实现了在传输链路抖动状态下的云网络数据的高效传输。然而该算法对数据传输过程中的流量控制不够,当数据流量出现异常增加时,算法将出现严重的存储性能下降的现象。陈杰等人<sup>[4]</sup>提出了一种数据特征映射机制的云网络数据存储算法,通过对过往存储的数据特征进行数据挖掘,实现了数据传输过程中的稳定性自适应匹配,提高了数据存储的效率。然而该算法需要通过数据挖掘机制对过往数据特征进行挖掘,当数据库处于动态更新或数据容量巨大时,将由于数据挖掘性能的下落而导致数据存储过程出现严重的下降现象。

鉴于当前技术实现过程中存在的一些不足,提出了一种基于海量存储云调度机制的云网络数据超带宽存储算法,通过对云网络存储及传输数据进行海量云调度转存,且结合数据存储中使用强度,最小传输粒度等数字特征,提取数据传输存储指纹梯度,并将该指纹梯度作为传输存储的指标,实现了海量存储过程中的云精确调度。仿真实验证明了本文算法的有效性。

## 2 海量存储云模型

由于云存储过程中数据均以离散状态进行存储<sup>[5]</sup>,且通过高速传输专线进行数据转存交互,因此需要对整个云网络架构做如下规定<sup>[6]</sup>:

1) 整个数据存储过程中均采用多跳方式实现数据传输,一旦中继节点出现瘫痪,则整个数据存储将处于终止状态<sup>[7]</sup>;

2) 数据流动具有透明特性,即最终存储节点并不关注中间传输链路,整个中间链路集合呈现云特性<sup>[8]</sup>;

3) 整个网络虽不存在固定的控制中心,但是存在中央数据库节点,任意数据传输及存储过程均需通过对该数据库的查询实现资源的调度<sup>[9]</sup>;

4) 任意一中继节点失效,均可通过其他的中继节点进行数据存储接力<sup>[9]</sup>。

由于对任意中继节点而言,进行数据传输过程即是对一定的带宽进行转存,数据汇聚成本可由如下的模型决定<sup>[10]</sup>:

$$E_{\text{sent}}(B) = B \sum_{t=0}^{B \rightarrow L} \int f_{\text{sent}}(t) dt \quad (1)$$

式中:  $E_{\text{sent}}(B)$  表示数据汇聚成本,  $B$  为数据存储量,  $f_{\text{sent}}(t)$  表示数据传输过程中的映射函数。

当传输过程满足多跳特征时,式(1)所示的数据汇聚成本模型将变为:

$$E_{\text{recv}}(B_1) = \sum_M \int_L B_1 f_{\text{sent}}(t) dt \quad (2)$$

式中:  $E_{\text{recv}}(B_1)$  为下一跳节点的数据汇聚成本,  $B_1$  为当前节点中数据贮存量,  $f_{\text{recv}}(t)$  表示数据传输过程中的映射函数。

由式(1)、(2)可知,对于任意一条数据传输链路而言,整体数据汇聚成本主要由当前节点数据存储量以及下一跳节点的汇聚成本决定,且由式(2)可进一步得知:传输过程中需要根据各条数据汇聚链路进行数据存储,因此其代价将随数据汇聚链路数量的增多而随之提高。

## 3 云网络数据超带宽存储算法

当存储节点进行数据存储过程时,若未将数据存储中使用强度,最小传输粒度等数字特征作为重要的数据存储参数,则将不得不随机选择一条性能较好的链路进行数据传输,难以提高数据存储的总容量,因此本文考虑到这些必要参数,对数据存储过程进行控制,以便能够较好的提

高数据存储带宽,改善海量数据存储效果<sup>[10]</sup>。整个算法由超宽带存储调度和梯度质量控制两部分构成。

1) 依据数据存储使用强度  $T(x)$  和带宽占用概率  $P(x)$  进行超宽带存储控制

由于数据存储在云网络中的各个独立存储节点上,不同的节点数据使用强度均有所不同,可以由数据存储使用强度  $T(x)$  和带宽占用概率  $P(x)$  来进行传输匹配。独立存储节点  $t_0$  上进行实时数据流接收,数据流服从指数为  $\lambda$  的泊松分布,且数据存储使用强度  $P(x)$  满足相同指数的分布。故数据存储使用强度  $T(x)$  的一阶矩  $E[T(x)]$  满足:

$$E[T(x)] = \frac{\lambda - E | T(x) |}{\lambda^2 E | T(x)^2 | - E | T(x) | - e} \quad (3)$$

式中:  $E | T(x) |$  和  $E | T(x)^2 |$  分别为  $T(x)$  的期望及二阶期望。

数据存储使用强度  $P(x)$  的一阶矩  $E[P(x)]$  与  $T(x)$  的一阶矩  $E[T(x)]$  满足线性反比例关系,即:

$$E[P(x)] = \frac{\lambda^2 E | T(x)^2 | - E | T(x) | - e}{\lambda - E | T(x) |} \quad (4)$$

由式(3)获取的一阶矩若为正数,则说明整个数据存储使用为正向关系,即数据存储能够通过链路进行超带宽传输;若获取的一阶矩为负数,说明需要采用一定的机制,对数据的粒度进行调整,以便能够加强数据传输强度,实现质量控制。

2) 按照最小传输粒度  $p'$  进行超带宽梯度传输

考虑到最小传输粒度的触发条件为式(3)获取的一阶矩符号,当仅当一阶矩符号为负数时,才需要对数据粒度进行调整,并通过该粒度对数据存储使用强度  $T(x)$  进行超带宽控制。

$T(x)$  与  $p'$  呈现负相关关系,因此当最小传输粒度被触发时,  $T(x)$  符号在传输周期内将均不发生变化,即:

$$T(x) \Rightarrow p' \quad (5)$$

式中:  $T(x) \Rightarrow p'$  的一阶矩  $E[T(x) \Rightarrow p']$  满足平稳随机莱分布,该分布的数学特征函数满足:

$$E[T(x) \Rightarrow p'] = \int E[T(x)] \quad (6)$$

可设下一时刻的数据最小传输粒度仍然为  $p'$ ,则最小传输粒度被激发时  $T(x)$  满足:

$$T(x) = \int E[T(x)] + E[T(x) \Rightarrow p'] \quad (7)$$

考虑到激发最小传输粒度时,网络中全部可用于调度的传输带宽处于有限使用状态<sup>[11-12]</sup>,且存储梯度  $\Delta$  能够对覆盖  $T(x)$  的全适应;因此对于任意时刻  $\Delta t$ ,梯度弹性系数  $\Delta T(x)$  满足:

$$\Delta T(x) = p' \int_{\Delta}^{\Delta} \sqrt{\Delta^2 - T(x)^2} dx \quad (8)$$

式中:  $p'$  为最小传输粒度,  $T(x)$  为数据存储使用强度。

据此可以获取存储指数  $\Delta \lambda$  满足:

$$\Delta\lambda = \frac{\mathbf{E} | T(x) | - \lambda}{1 - \lambda^2 \mathbf{E} | T(x)^2 | - \mathbf{E} | T(x) |} p' \times \int_{\Delta}^{\Delta} \sqrt{\Delta^2 - T(x)^2} dx \quad (9)$$

相应的,梯度弹性系数  $\Delta$  与  $T(x)$  需满足如下条件,则进行二次存储传输,如图 1 所示。

$$T(x) = \Delta(1 - \Delta)(\sqrt[3]{1 - \Delta\lambda}) \quad (10)$$

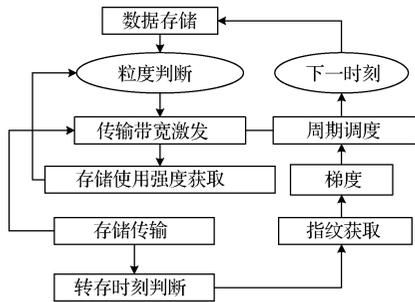


图 1 本文云网络数据超带宽存储算法过程

#### 4 仿真实验

为评估本文算法存储带宽、存储时延的性能,采用当前常用的超线性存储调度算法(super linear memory scheduling algorithm, SLMS)<sup>[13]</sup>与本文算法进行测试。实验仿真参数如表 1 所示。

表 1 仿真参数表

参数	数值
云网络区域/km <sup>2</sup>	36×72
网络仿真时间/h	48
转存节点密度	不低于 24
传输速率/bps	不低于 2 M
节点功率	不低于 48 J/dB
比特长度 $m$	1024 bit

##### 1) 存储带宽

图 2 所示显示了在节点传输速率不断增大的情况下,本文算法与 SLMS 算法的存储带宽测试,由图可知随着传输速率的不断增大,本文算法的存储带宽始终要远高于 SLMS 算法,这是由于本文算法采用梯度转存的方式,能够随着数据传输速率的不断增大,降低网络中因数据传输量扩大而导致的带宽瓶颈效应,且可用的存储节点数目也要高于对照组算法,因此本文算法存储带宽较对照组算法具有明显的优势,特别是在高传输速率时,由于数据传输过程中存在梯度关系,本文算法能够将不同梯度的数据进行迭次转存,因而增强了转存过程中的鲁棒性,使得转存

率也随之上升,从而提高了存储带宽;SLMS 算法由于仅仅采用最佳链路传输模式,当网络中数据随着传输速率的增大而增大时,将由于数据流的增加而发生拥塞现象,因而使得数据存储带宽要小于本文算法。

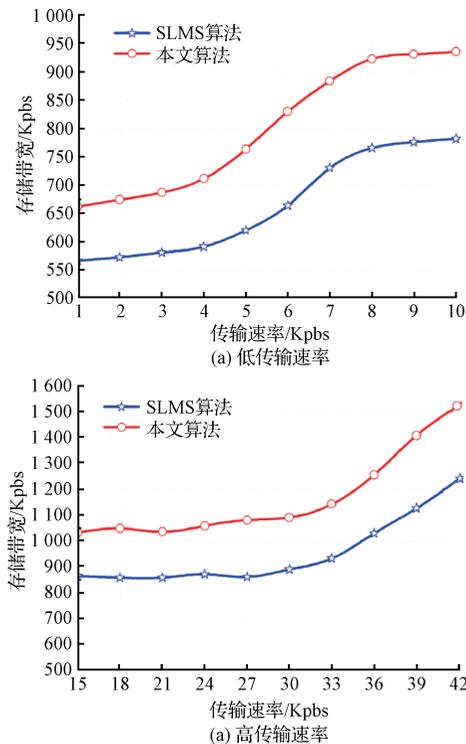


图 2 不同传输速率条件下,各算法的存储带宽测试

##### 2) 存储时延

图 3 所示显示了在节点功率不断提高的情况下,本文算法与对照组算法存储时延的测试结果,由图可知,随着节点功率的不断增大,本文算法与对照组算法的存储时延都呈现扩大的趋势,然而本文算法的存储时延增加幅度有限,波动性较低,且始终要低于对照组算法。这是由于综合考虑了数据存储中使用强度,最小传输粒度等数字特征,能够在节点功率不断增大的情况下进一步减少数据拥塞情况的发生,起到了存储分流的作用,因而存储时延上升幅度有限。而对照组算法由于采用简单调度机制,一旦由于节点功率增大而带来的数据流量突增的情况发生,则由于发生严重的数据拥塞而导致传输过载现象发生,从而提高而来存储时延水平。此外,当网络节点功率处于较高水平时,由于节点功率的迅速上升,导致网络节点性能也随之下降,由于本文算法采用梯度转存方式,能够在网络节点性能下降时降低转存链路的抖动行,因而本文算法的存储时延指标具有明显优势。

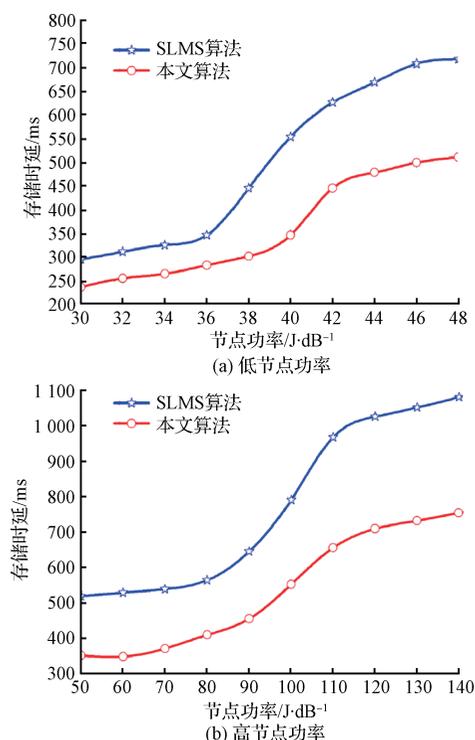


图3 不同传输速率条件下,各算法的存储时延测试

## 5 结论

为解决当前云网络数据存储研究中存在的数据存储效率不高,且转存困难等难题,提出了一种基于海量存储云调度机制的云网络数据超带宽存储算法。通过综合考虑数据存储中使用强度,最小传输粒度等数字特征,并结合数据转存梯度的方式,大大提高了数据存储的效率,且具有时延较低的特性。仿真实验证明了本文算法的有效性。

下一步,将采用球形传输一体化方式,通过提高各个节点存储链路的正交性能,降低因数据存储中使用强度,最小传输粒度等数字特征的相似程度而导致的重复传输现象,进一步提高本文算法的使用价值。

### 参考文献

[1] 黄庆卿, 汤宝平, 邓蕾. 无线传感网络数据分块无损压缩方法[J]. 仪器仪表学报, 2015, 36(7): 1605-1610.

- [2] 李彦, 李镛. 基于嵌入式 Linux 系统的双网卡大数据传输[J]. 电子测量与仪器学报, 2014, 28(9): 1027-1032.
- [3] 沈晴霓, 杨雅辉, 禹熹, 等. 一种面向多租户云存储平台的访问控制策略[J]. 小型微型计算机系统, 2011, 11(123): 2223-2229.
- [4] 陈杰. 大数据场景下的云存储技术与应用[J]. 中兴通讯技术, 2012, 6(9): 47-51.
- [5] 陈硕章, 刘海斌, 王春. 基于 LabVIEW 和云数据库存储的远程测试系统设[J]. 国外电子测量技术, 2016, 35(6): 67-72.
- [6] 傅颖勋, 罗圣美, 舒继武. 安全云存储系统与关键技术综述[J]. 计算机研究与发展, 2013, 14(1): 136-145.
- [7] TAN SH, JIA Y. NaEPASC: A novel and efficient public auditing scheme for cloud data[J]. Frontiers of Information Technology & Electronic Engineering, 2014, 15(9): 794-804.
- [8] 徐小龙. 一种基于节点负载的云存储平台海量数据转存算法[J]. 计算机科学, 2013, 02(9): 98-102.
- [9] 陆晓玮, 陈健. 基于深度包检测技术的邮件过滤研究[J]. 电子测量技术, 2014, 37(4): 25-28.
- [10] 罗红, 杨放春, 刘永和. 云网络中分布式数据汇聚路由算法[J]. 北京邮电大学学报, 2012, 30(1): 9-13
- [11] AHLSSWEDE R, CAI N, YEUNG R W. Clouds Network information flow [J]. IEEE Trans. on Information Theory, 2014, 46(4): 1204-1216.
- [12] NIU B. A cooperation stimulation strategy in Clouds networks [J]. IEEE Transactions on Signal Processing, 2011, 59(5): 2355-2369.
- [13] XU P, CHEN H, ZOU D, et al. Fine-grained and heterogeneous proxy re-encryption for secure cloud storage [J]. Science Bulletin, 2014, 59(32): 4201-4209.

### 作者简介

钱伟强, 1982年2月出生, 硕士, 讲师, 研究方向为计算机网络, 计算机应用、数据库系统。  
E-mail: QianWqiang1982e@139.com