基于多目标优化的云存储副本分布策略的研究

张华伟[1] 李志华[1,2]

(江南大学物联网应用技术教育部工程研究中心 无锡 214122)[1]

(江南大学物联网工程学院轻工过程先进控制教育部重点实验室 无锡 214122)[2]

摘 要 针对现有云存储副本分布策略优化目标比较单一的不足,提出了局部最佳分布策略(Local Optimum Distribution,LODS)。LODS策略通过给出一系列新定义并利用一致性哈希函数来缩小副本分布的节点选择范围,进一步结合层次分析法,将一定决策半径内的节点作为方案层中的候选对象,通过更深入地研究云存储多目标优化准则对其优化从而最终选择出当前候选方案中的最佳目标节点。实验结果表明,通过优化的最优决策半径取值相对稳定,不随云存储系统规模的扩展和数据的增多而剧烈变化,并且当取值最佳决策半径时,LODS策略的存储负载平衡、热度负载平衡、等待时间性能高于HDFS、Amazon S3等系统中所采用的副本分布策略。

关键词 云存储,一致性哈希,层次分析法,副本分布,多目标优化

中图法分类号 TP312 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.4.007

Research on Data Distribution Strategy in Cloud Storage System Based on Multi-objective Optimization

ZHANG Hua-wei[1] LI Zhi-hua[1,2]

(Engineering Research Center of IoT Technology Application Ministry of Education, Jiangnan University, Wuxi 214122, China)[1]

(Key Laboratory of Advanced Process Control for Light Industry of Ministry of Education, School of IoT Engineering, Jiangnan University, Wuxi 214122, China)[2]

Abstract The data distribution strategy in cloud storage systems has shortcomings that the distribution strategy only considers the load balancing among physical storage nodes instead of the heat load balancing, waiting time and other factors. Aiming at this question, a rational and efficient data distribution strategy local optimum distribution (LODS) was proposed. By defining a series of new definitions to describe the correlations in LODS, employing the hashing method to act as the core ideal and presenting a couple of multi-objective optimization rules, LODS combines the above innovations and the AHP method to give the finally optimized storage scheme. The proposed storage scheme can search the target storage node in cloud system within a local range with a optimized decision radius. The decision radius is stable without variation while the large-scale storage amount is tested. Once the decision radius is "0", what the LODS degenerates is almost corresponding to that of Amazon S3. Experiments show that the average advantage of the proposed cloud storage strategy overs that used in HDFS, Amazon S3 in terms of the strategies of storage load balancing, load balancing heat and waiting time.

Keywords Cloud storage, Consistent hashing, AHP, Data distribution, Multi-objective optimization

1 引言

云存储系统中的副本分布策略主要研究如何将数据对象的副本高效地分布到云存储系统中去,从而满足云存储系统的存储节点间的负载平衡和较高的用户服务质量的要求。当前副本分布策略主要包括两类:一类是以HDFS[1]、GFS[2]为代表的基于集中式的存储目录来定位数据对象的存储位置的副本分布策略;另一类是以Amazon S3[3]为代表的基于一致性哈希[4]的副本分布策略。这两种副本分布策略的默认副本数量都为3。HDFS策略的主要思想是:两个副本存放在本地机架的不同存储节点上,这样既可以保证用户与云存储节点之间快速的数据传输,又可以在当其中一个副本上的数据块遭到破坏时,快速地从本机架获取数据块的备份,第3个副本放置在其它机架上,当本地机架遭遇灾难性损坏时可以从其它位置的机架上恢复本地机架中的数据。但是HDFS的副本分布策略存在的主要不足是:选择的是可用的副本存储节点而不是最佳的副本存储节点。Amazon S3策略的主要思想是:采用一致性哈希将一个数据块的3个副本随机地分布到各存储节点上。Amazon S3策略的算法一次决策的时间复杂度为O(1),决策时间较短。但是副本分布的随机性过强,一个数据块的3个副本分布可能会出现以下两种不理想的情况:(1)3个副本都选择了异地机架的存储节点存储,本地向

到稿日期:2014-05-09 返修日期:2014-09-21 本文受江苏省科技厅产学研前瞻项目(BY2013015-23),中央高校科研专项(JUSRP211A41)资助。

张华伟(1988-),男,硕士生,主要研究方向为云计算、网格与分布式计算,E-mail:zhhw-2009@163.com;李志华(1969-),男,博士,副教授,主要研究方向为网络技术、物联网技术、信息与网络安全、物联网安全技术等。

异地传输数据的速度会比较慢；(2)3 个副本都分布在同一机架甚至同一存储节点上，当一个存储节点或者一个机架发生崩溃时，为数据块的恢复显著地增加了难度。

对于 HDFS 和 Amazon S3 副本分布策略中存在的不足，文献[6-9]针对存储负载平衡方面进行了优化，文献[10]针对热度负载平衡方面进行了优化，文献[11,12]针对用户服务质量(QoS)方面进行了优化，文献[13]针对系统节能方面进行了优化，文献[14,15]针对副本之间的关联性进行优化。但这些优化都是针对单一目标进行的，没有从整体性能上进行考虑。

面向上述文献针对单一目标优化的不足，本文提出了以一致性哈希和层次分析法[5]为基础的局部最佳分布策略(Local Optimum Distribution Strategy，LODS)。一致性哈希可以使副本总体分布相对均匀，再借助层次分析法对副本局部范围内的节点选择进行多目标优化，从而使其在局部节点选择时达到最优，较好地克服了上述目标优化方案的不足。在此基础上，进一步提出了副本分布策略综合性评价方法和评价指标，很好地弥补了当前的评价方法、评价标准不适用于局部优化过程评价的不足。

2 云存储局部最佳分布策略

2.1 LODS 的基本思想

为了表达方便，在此首先给出如下新定义，如图 1 所示。

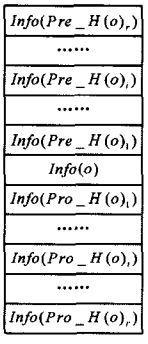

图 1 决策表结构

定义 1(决策中心) 从数据对象映射到环形哈希数值空间中的数值点出发，以顺时针寻找到的第一个由存储节点映射到的数值作为决策中心，用符号 o 表示。

为了方便叙述，数值 o 对应的存储节点也用 o 表示。

定义 2(决策半径) 从决策中心出发顺时针和逆时针寻找相同个数的由存储节点映射到环形哈希数值空间中的数值，沿一个方向需要寻找的个数称为决策半径，符号表示为 r。

如果数值 key 为从决策中心出发顺时针或者逆时针寻找到的第 i 个由存储节点映射到的数值，则称 key 与决策中心 o 的距离为 i，顺时针与决策中心 o 距离 i 的数值记为 $Pre(o)_i$，同样逆时针与决策中心 o 距离 i 的数值记为 $Pro(o)_i$，与它们对应的存储节点分别记为 $Pre\_H(o)_i$ 和 $Pro\_H(o)_i$。

定义 3(决策域) 定义

$Pre\_H(o)=\{Pre\_H(o)_1, Pre\_H(o)_2, \cdots, Pre\_H(o)_r\}$ 为决策中心 o 的向前决策域。

同理，定义

$Pro\_H(o)=\{Pro\_H(o)_1, Pro\_H(o)_2, \cdots, Pro\_H(o)_r\}$ 为决策中心 o 的向后决策域。

$P\_H(o)=Pre\_H(o)\cup Pro\_H(o)\cup\{o\}$ 为决策中心 o 的决策域。

定义 4(决策表) 记录 $P\_H(o)$ 中所有存储节点状态的表称为决策中心 o 的决策表，结构如图 1 所示，其中 $Info(o)$ 表示存储节点 o 的状态信息。

定义 5(有效决策集) 假设云存储系统中的同一数据块对应的副本数为 3，则决策域中可以参与决策的存储节点需要满足以下条件：

(1)有足够的空间存储数据块的一个副本；

(2)该存储节点存储的该数据块的副本数小于等于 0；

(3)该存储节点所在机架存储的该数据块的副本数小于等于 1；

(4)该存储节点等待请求数低于该节点的上限。

定义以上条件称为 Λ 条件，则有

$P\_U(o)=\{S|S\in P\_H(o), S$ 满足 Λ 条件$\}$

为决策中心 o 的有效决策集。

定义 6(有效决策集的一次扩展) 若 $P\_U(o)=\emptyset$，即当前决策域内没有适合存储当前副本的存储节点时，需要从 $Pre\_H(Pre(o)_r)\cup Pro\_H(Pro(o)_r)$ 中选择满足 Λ 条件的存储节点加入有效决策集，同时 $P\_H(o)=P\_H(o)\cup Pre\_H(Pre(o)_r)\cup Pro\_H(Pro(o)_r)$，这一过程称为有效决策集的一次扩展。

图 2 同时展示了不同定义之间的关系或关联。

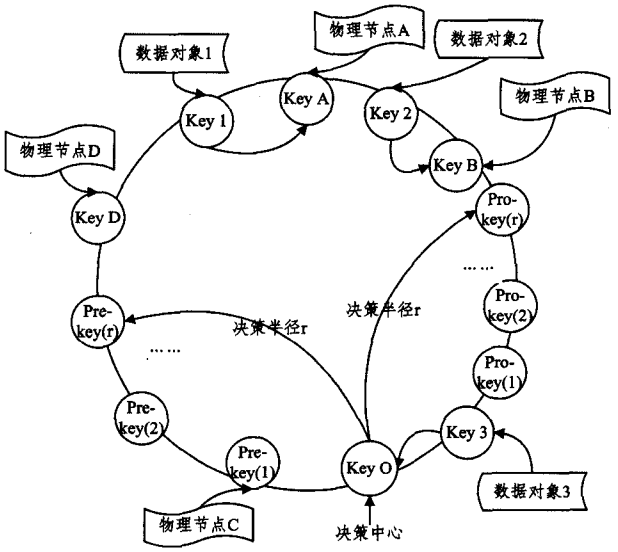

图 2 LODS 决策域定义

2.2 影响存储节点选择的因素分析

影响副本存储节点选择的因素主要包括：存储载后负载率、相对热度负载、预计等待时间。详细介绍如下：

(1)存储载后负载率

载后负载率主要为了衡量相同的副本数据块对于异构存储节点所造成的压力不同，即计算当前副本 $Repli_j$ 加载到节点 N_i 后 N_i 的负载率，如式(1)所示：

$$Load\_r(N_i)=\frac{Load(N_i)+Burden(Repli_j)}{Storage(N_i)} \quad (1)$$

其中，$Load(N_i)$ 表示 N_i 的当前负载，$Burden(Repli_j)$ 表示存储副本 $Repli_j$ 所需要的存储空间，$Storage(N_i)$ 表示 N_i 的存储能力，则 $Load\_r(N_i)$ 表示 N_i 的载后负载率。

(2)相对热度负载

用节点相对热度来表示节点接收到的副本读写请求的总

次数，由于云存储大多数遵循一次写入多次读取的方式，因此节点 N_i 的热度 $Hot(N_i)$ 可由式(2)表示：

$$Hot(N_i)=Sum(N_i)+\sum_{Repli_m\in N_i} Read\_num(Repli_m) \quad (2)$$

其中，$Sum(N_i)$ 表示 N_i 中副本的数量，$Read\_num(Repli_m)$ 表示副本 $Repli_m$ 请求读的次数。

假设 $Hot\_handle(N_i)$ 为节点 N_i 在单位时间处理的请求数，则节点 N_i 相对热度负载 $Re\_Hot(N_i)$ 可由式(3)表示：

$$Re\_Hot(N_i)=\frac{Hot(N_i)+1}{Hot\_handle(N_i)} \quad (3)$$

其中，$Hot\_handle(N_i)$ 表示 N_i 单位时间处理的请求数，+1 表示假设当前副本由 N_i 接收。

(3)预计等待时间

用预计等待时间来预先估计存储请求的等待时间。预计等待时间 $Pretict\_w(N_i)$ 可表示如下：

$$Pretict\_w(N_i)=Pretict\_t(N_i)+Trans\_t(N_i)-Arrive\_t(Repli_j) \quad (4)$$

其中，$Trans\_t(N_i)=\frac{Burden(Repli_j)}{Trans\_r(N_i,Repli_j)}$ 表示副本 $Repli_j$ 向节点 N_i 的传输时间，是副本 $Repli_j$ 对于节点的负担 $Burden(Repli_j)$ 与节点 N_i 同 $Repli_j$ 所在客户端之间传输速度的比值；$Pretict\_t(N_i)$ 表示节点 N_i 现有请求队列的完成时间，$Arrive\_t(Repli_j)$ 表示副本 $Repli_j$ 存储请求的到达时间。

(4)存储载后负载率、相对热度负载、预计等待时间的重要性分析

假设存储负载率、相对热度负载、预计等待时间的重要性分别为 p、q、v，三者的取值与决策域内各存储节点所处的状态有关。以存储负载率为例，在决策域内各存储节点总体的存储负载率较低的条件下，存储负载的不均衡对于决策域内各存储节点总体的性能影响较小，因此存储负载率的重要性 p 随决策域内各存储节点总体存储负载率提高而变大。同样，相对热度负载、预计等待时间的重要性也分别随决策域内各存储节点相对热度负载和预计等待时间的增大而增大。

与影响存储节点选择的因素相对应，决策域 $U(o)$ 所处的状态可用各存储节点总体存储负载率、总体相对热度负载和等待队列占有率 3 个指标衡量，如式(5)—式(7)所示。

决策域 $U(o)$ 内各存储节点总体存储负载率：

$$Load\_S(U(o))=\frac{\sum_{N_i\in U(o)} Load(N_i)}{\sum_{N_i\in U(o)} Storage(N_i)} \quad (5)$$

是决策域 $U(o)$ 中各存储节点负载之和与存储空间之和的比值。

决策域 $U(o)$ 内各存储节点总体相对热度负载：

$$Load\_H(U(o))=\frac{\sum_{N_i\in U(o)} Hot(N_i)}{\sum_{N_i\in U(o)} Hot\_handle(N_i)} \quad (6)$$

是决策域 $U(o)$ 中各存储节点热度负载之和与单位时间处理的请求数之和的比值。

决策域 $U(o)$ 内各存储节点总体的预计等待时间长短的程度不容易衡量，可近似用决策域内各存储节点的等待队列长度之和与允许最大等待队列长度之和表示，则等待队列占有率为：

$$Wait\_L(U(o))=\frac{\sum_{N_i\in U(o)} Wait\_o(N_i)}{\sum_{N_i\in U(o)} Wait\_m(N_i)} \quad (7)$$

其中，$Wait\_o(N_i)$ 表示存储节点 N_i 等待队列长度，$Wait\_m(N_i)$ 表示存储节点 N_i 等待队列允许的最大长度。

由式(5)—式(7)可知，$Load\_S(U(o))$、$Load\_H(U(o))$、$Wait\_L(U(o))$ 的取值范围均为[0,1]。为了使存储负载率、相对热度负载、预计等待时间三者的重要性对于决策域内各存储节点的总体状态敏感，p、q、v 表示如下：

$$p=p_0+a*\lfloor 10*Load\_S(U(o))\rfloor \quad (8)$$

$$q=q_0+b*\lfloor 10*Load\_H(U(o))\rfloor \quad (9)$$

$$v=v_0+c*\lfloor 10*Wait\_L(U(o))\rfloor \quad (10)$$

其中，p_0、q_0、v_0 为初始化的 3 种因素的重要性，a、b、c 为大于等于 0 的整数。特殊地，当 p_0、q_0、v_0、a、b、c 均为 1 时，p、q、v 在[1,10]范围内的整数间变化。

(5)云存储系统状态信息的收集和维护

计算决策域内各存储节点的存储载后负载率、相对热度负载、预计等待时间需要收集各存储节点的状态信息，包括节点的存储能力、节点的当前存储负载、节点的热度负载、节点的热度处理能力、用户到存储节点的数据传输速率等信息。前四者的信息可以直接从各存储节点中获取，但是用户到存储节点的数据传输速率复杂而多变，难以测量，可以通过以下方式近似获取。用户向存储节点发送数据块经过的网络分为两大部分，数据中心内部网络和外部公用网络。数据中心内部采用高速光纤，所以我们假设传输速度的“瓶颈”在外部公用网络，进而数据块传输速率近似于数据块在外部网络传输到数据中心的速率。为了方便衡量某一用户到数据中心的传输速率，可以将云存储系统的服务区域分块，如图 3 所示，区域 5 中的测试机固定时间间隔向各数据中心发送测试数据块来计算测试机到数据中心的数据传输速率，用公式

$$\frac{\text{测试数据传输速率}}{\text{测试机带宽}}\times\text{用户带宽}$$

计算区域 5 中各用户到数据中心各存储节点传输速率的近似值。

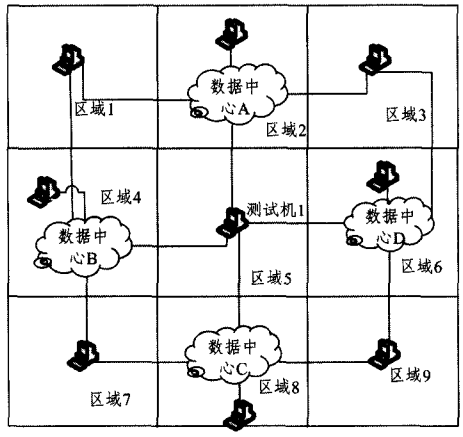

图 3 服务区域分块示意图

决策中心将收集到的决策域内各存储节点的存储能力、当前存储负载、热度负载、热度处理能力等状态信息记录到决策表中，当某一存储节点的状态信息发生变化时，将会更新以此存储节点为决策中心的决策域内所有存储节点的决策表。区域测试机到各数据中的数据传输速率与测试机带宽的比值记录在目录节点上，定时更新。

2.3 层次分析模型

层次分析法是一种将定性与定量分析方法相结合的多目

标决策分析方法。该方法的主要思想是通过将复杂问题分解为若干层次及若干因素，对两两指标之间的重要程度做出比较和判断。该方法通过建立判断矩阵，并计算判断矩阵的最大特征值以及对应特征向量，从而得出不同方案重要性程度的权重。在此，权重是作为最佳方案选择的依据。层次分析模型由 3 个层次组成，至上而下分别是：目标层、准则层和方案层。准则层根据问题的复杂性需要还可以分成多个层次。

在有效决策域中，可根据影响存储节点选择的因素来选择最适合副本存储的存储节点。把这一过程转化为图 4 所示的层次分析模型。

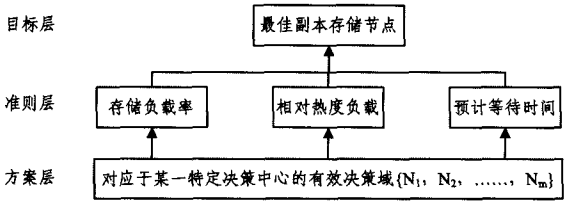

图 4　局部最优的层次分析模型

基于图 4 的层次分析模型需要构造 4 个成对比较矩阵：准则层 3 个要素之间的相对比较矩阵，以及有效决策域集合中各元素之间针对准则层 3 个要素的成对比较矩阵 B_1、B_2、B_3。

根据上文所提到的存储负载率、相对热度负载、预计等待时间的重要性分别为 p、q、v，可以得到成对比较矩阵：

$$A=\begin{pmatrix} 1 & p/q & p/v \\ q/p & 1 & q/v \\ v/p & v/q & 1 \end{pmatrix} \tag{11}$$

显然 A 为一致性矩阵，A 的每一列都是其特征向量，本文采用和法[16]求得矩阵 A 的单位特征向量：

$$A=\begin{pmatrix} 1 & p/q & p/v \\ q/p & 1 & q/v \\ v/p & v/q & 1 \end{pmatrix} \xrightarrow{\text{求行和}} \begin{pmatrix} \frac{qv+pv+pq}{qv} \\ \frac{qv+pv+pq}{pv} \\ \frac{qv+pv+pq}{pq} \end{pmatrix}$$

$$\xrightarrow{\text{归一化}} \begin{pmatrix} \frac{p(qv+pv+pq)}{p^2(v+q)+q^2(p+v)+v^2(p+q)} \\ \frac{q(qv+pv+pq)}{p^2(v+q)+q^2(p+v)+v^2(p+q)} \\ \frac{v(qv+pv+pq)}{p^2(v+q)+q^2(p+v)+v^2(p+q)} \end{pmatrix}$$

$$=\begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}=W \tag{12}$$

成对比较矩阵 B_1、B_2、B_3 如式(13)—式(15)所示。

$$B_1=\begin{pmatrix} 1 & \frac{Load\_r(N_1)}{Load\_r(N_2)} & \cdots & \frac{Load\_r(N_1)}{Load\_r(N_m)} \\ \frac{Load\_r(N_2)}{Load\_r(N_1)} & 1 & \cdots & \frac{Load\_r(N_2)}{Load\_r(N_m)} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{Load\_r(N_m)}{Load\_r(N_1)} & \frac{Load\_r(N_m)}{Load\_r(N_2)} & \cdots & 1 \end{pmatrix} \tag{13}$$

$$B_2=\begin{pmatrix} 1 & \frac{Hot(N_1)}{Hot(N_2)} & \cdots & \frac{Hot(N_1)}{Hot(N_m)} \\ \frac{Hot(N_2)}{Hot(N_1)} & 1 & \cdots & \frac{Hot(N_2)}{Hot(N_m)} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{Hot(N_1)}{Hot(N_m)} & \frac{Hot(N_2)}{Hot(N_m)} & \cdots & 1 \end{pmatrix} \tag{14}$$

$$B_3=\begin{pmatrix} 1 & \frac{Pretict\_w(N_1)}{Pretict\_w(N_2)} & \cdots & \frac{Pretict\_w(N_1)}{Pretict\_w(N_m)} \\ \frac{Pretict\_w(N_2)}{Pretict\_w(N_1)} & 1 & \cdots & \frac{Pretict\_w(N_2)}{Pretict\_w(N_m)} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{Pretict\_w(N_m)}{Pretict\_w(N_1)} & \frac{Pretict\_w(N_m)}{Pretict\_w(N_2)} & \cdots & 1 \end{pmatrix} \tag{15}$$

假设 B_1、B_2、B_3 的特征向量分别为：

$W_1=\{w_{11},w_{12},\cdots,w_{1m}\}$

$W_2=\{w_{21},w_{22},\cdots,w_{2m}\}$

$W_3=\{w_{31},w_{32},\cdots,w_{3m}\}$

则有效决策域中的存储节点 N_i 对目标的组合权重如式(16)所示。

$$U_i=\sum_{j=1}^{j\leqslant 3} w_j * w_{ij}, i=1,2,\cdots,m \tag{16}$$

由于 3 种因素都是目标选择的负因素，选择有效决策域中组合权重最小的存储节点作为最终的优化目标。

2.4　副本分布算法

综上所述，副本分布策略算法概括如下。

LODS 算法

Step1　系统初始化，将云存储系统的所有存储节点映射到环形哈希数值空间，初始化决策半径 r 和各个存储节点的决策表；

Step2　当目录节点接收到副本数据对象 $Repli_j$ 的存储请求时，由定义 1 寻找与 $Repli_j$ 相对应的决策中心 o，将 $Repli_j$ 信息和请求用户到各数据中心的网络信息发送到决策中心 o；

Step3　决策中心 o 根据定义 5 决策半径 r 构造决策中心 o 的有效决策集 $P\_U(o)$；

Step4　如果 $P\_U(o)=\emptyset$ 并且决策域范围小于云存储系统范围，执行 Step5，否则执行 Step6；

Step5　对有效决策集进行一次扩展，返回 Step4；

Step6　如果 $P\_U(o)=\emptyset$，无合适存储节点，返回错误，如果 $P\_U(o)$ 只有一个元素，该元素对应的存储节点为最佳存储节点 N，执行 Step9，否则执行 Step7；

Step7　对 $P\_U(o)$ 中的每一个存储节点，分别用式(1)、式(3)、式(4)计算存储载后负载率、相对热度负载、预计等待时间；

Step8　根据式(5)—式(11)针对有效决策集 $P\_U(o)$ 对应的决策域 $P\_H(o)$ 当前所处的状态构造成对比较矩阵 A。

Step9　以 $P\_U(o)$ 中所有元素作为方案层中的候选方案，以元素节点的存储载后负载率、相对热度负载、预计等待时间作为准则层中的决策准则，选择最佳存储节点 N，并将决策结果返回到目录节点；

Step10　将 $Repli_j$ 请求加入 N 的请求等待队列；

Step11　更新决策域中所有节点的决策表，返回成功；

LODS算法的时间复杂度主要集中在Step9和Step11，两者的时间复杂度都为$O(r^2)$，r为决策半径，其他步骤的时间复杂度均不大于$O(n)$，n为云存储系统的节点数量，因此算法的时间复杂度为$O(r^2)$。然而，Step3－Step11是分散到各决策中心上执行的，所以算法的效率提升λ_n倍。另外，成对比较矩阵A随云存储系统的状态变化而改变，优先优化云存储系统的紧缺资源。

2.5 存储节点的增加与删除

云存储系统动态地添加和删除存储节点是十分正常的，LODS副本分布算法在存储节点增加或删除时做出以下处理。

(1)存储节点的增加

存储节点S从图5所示位置加入云存储系统，假设从S出发顺时针寻找到的第一个存储节点为o，更新o的决策域$P_H(o)$中所有存储节点的决策表相应位置的信息。

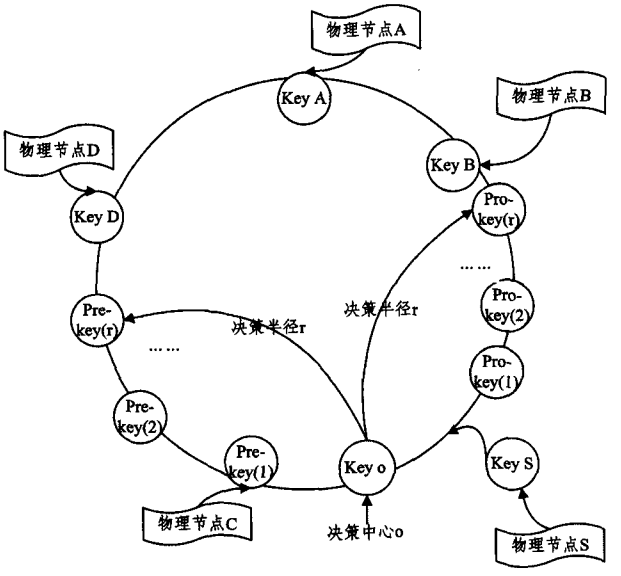

图5 存储节点的增加

(2)存储节点的删除

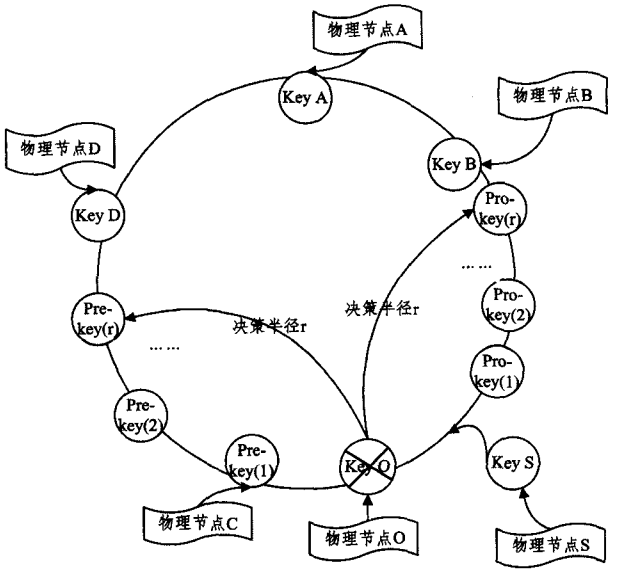

图6 存储节点的删除

假设存储节点o删除，如图6所示，从o的决策域$P_H(o)$中选择与o同机架的存储节点，如果不存在则选择所属机架与o所属机架最近的存储节点，将o上的数据块分散到这些存储节点上，并更新o的决策域$P_H(o)$中所有存储节点的决策表相应位置的信息。

3 副本分布策略的性能评价

本文的副本分布策略综合考虑了存储负载率、相对热度负载、平均等待时间、决策时间4个要素。现有的只考虑节点负载平衡的系统性能的评价方式不适合对本文策略的评价。因此，本文进一步提出了新的评价指标。

3.1 存储负载平衡

存储负载平衡表示副本在系统分布的均衡程度，可以通过式(17)来衡量：

$$S_{load}=\frac{1}{n}\sum_{N_i\in System}|Load\_r(N_i)-E_{load}(System)| \quad (17)$$

其中，n为云存储系统$System$中存储节点的数量，$Load_r(N_i)$为云存储节点N_i的存储负载率，$E_{load}(System)$表示云存储系统的整体存储负载率，是系统中副本所占存储容量之和与系统总存储容量的比值。S_{load}表示了各节点负载率偏离系统总体负载率的程度。S_{load}越大，均衡性越差，系统的性能越差，采用的副本分布策略在存储负载均衡化方面表现也越差。

3.2 热度负载平衡

热度负载平衡表示云存储用户访问热度在云存储系统各节点分布的平衡度量，与存储负载平衡的评价方式类似，用式(18)表示：

$$S_{hot}=\frac{1}{n}\sum_{N_i\in System}|Re_{hot}(N_i)-E_{hot}(System)| \quad (18)$$

其中，$Re_{hot}(N_i)$表示节点N_i的访问热度，$E_{hot}(System)$表示云存储系统的相对热度，是系统中各节点的访问热度之和与系统总处理能力的比值。

3.3 副本存储请求的平均等待时间

副本存储请求的平均等待时间是副本存储请求的等待时间之和与副本存储请求数的比值，计算如式(19)所示：

$$Aver\_w=\frac{1}{m}\sum_{N_i\in System}Wait\_w(N_i) \quad (19)$$

其中，$Wait_w(N_i)$表示节点N_i存储请求的总等待时间，m为系统接收的存储请求的总量。

3.4 决策时间

由于LODS算法的时间复杂度为$O(r^2)$，在考虑30万条决策需求的情况下，利用曲线拟合的方法得到30万条决策需求的时间总和与决策半径r的关系如式(20)所示，关系图如图7所示。

$$time=25.6549*r^2+136.347*r+18.5515 \quad (20)$$

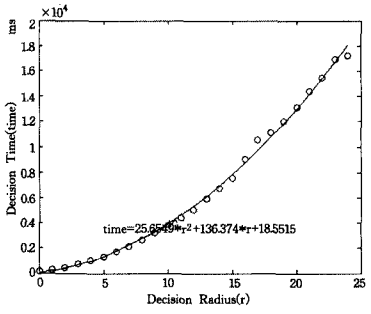

图7 决策时间与决策半径的关系

总之，本节新提出的评价指标并非是评价某一单一副本分布策略的好坏，而是对一组待评价的副本按分布策略进行优劣排序。针对存储负载率、相对热度负载、平均等待时间、决策时间 4 种因素分别计算各方案的组合权重，组合权重越小，方案的性能越优。

4 实验结果分析

为了验证本文所提出的副本分布策略的可行性和综合性能，用 java 语言编写了模拟云存储系统对用户存储请求的服务过程。存储节点主机是根据最近 10 年主流的主机配置随机产生的，数据块是随机产生的小于等于 64M、大于 3M 的块[17]。

在云存储系统中包含 5 个数据中心和 9000 个用户，每个数据中心对应一个机架。5 个数据中心分别包含 300、250、200、150、100 个存储节点。10000 个用户平均分成 10 个区域，随机存储 90 万个数据块。10 个区域的测试主机的网络带宽均设为 4M。10000 个用户的网络带宽在 2M、4M、8M、20M 中随机产生。9 个区域到 5 个数据中心的数据传输速率在 100kB～400kB 区间动态变化。在只考虑存储负载平衡、热度负载平衡、副本存储请求的平均等待时间的条件下，假设初始化三者的重要性相同均为 1，则成对比较矩阵为：

$$A=\begin{pmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{pmatrix}$$

设定式(8)—式(10)中的参数 a、b、c 均为 1，进而可以得到决策半径与组合权重的关系如图 8 所示。

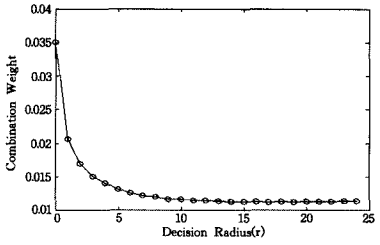

图 8 不考虑决策时间条件下的组合权重与决策半径的关系

不难看出当决策半径从 1 增加到 9 时，系统的组合权重迅速下降，然后稳定在一个较低的水平，这说明系统性能在决策半径从 1 增加到 9 时迅速提高，然后稳定在一个较高的水平。

从图 7 可以看出，决策时间开销在决策半径增加时迅速增加。把决策时间开销加入评价范围时，可以得到决策半径与组合权重的关系，如图 9 所示。

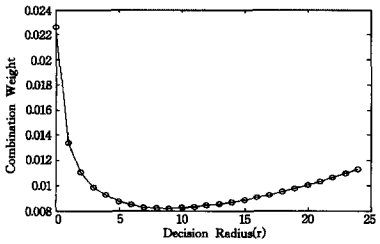

图 9 考虑决策时间条件下的决策半径与组合权重关系图

决策半径从 0 增加到 9 时，组合权重不断减小，当决策半径再次增加时，组合权重不断增大，在决策半径为 9 时，组合权重最小。这说明云存储系统的决策半径为 9 时最佳。

当节点数目和数据块数目分别取表 1 中的值时，可以得到决策半径与组合权重的关系，如图 10 所示，并将不同条件的最佳决策半径记录在表 1 中。可以看出，在不同节点数目和数据块数目条件下，组合权重随决策半径的变化趋势和取值大体一致，且最佳决策半径集中在 8 和 9 两种取值上，并且这两种取值下组合权重相差很小。

表 1

节点编号	data1	data2	data3	data4	data5	data6	data7	data8
节点数目(千)	0.5	0.5	1	1	1	1	1.5	1.5
数据块数目(万)	9	45	90	45	9	90	90	45
最佳决策半径	8	9	9	8	9	8	8	9

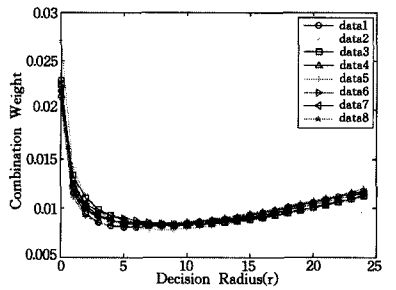

图 10 不同存储节点和数据块条件下的组合权重与决策半径的关系

进一步考虑一般情况，假设云存储系统的存储节点数目为 m，则当决策半径为 0 时，局部最佳副本分布策略会退化为 Amazon S3 的基于一致性哈希的副本分布策略。由图 11 可以看出，LODS 在存储负载率平衡性、热度负载平衡性、平均等待时间上优于 Amazon S3、HDFS 和文献[7]中的基于虚拟节点的副本分布策略(VID)。另外，LODS 的决策时间略高于 Amazon S3、HDFS 和 VID。LODS 副本分布算法分为两个主要部分，第一部分是将数据块副本映射到决策中心，第二部分是为数据块副本在此决策中心对应的决策域内寻找最佳的存储节点。由上述算法的分析可知，副本分布算法的时间复杂度主要集中在第二部分，为了保障副本分布算法的效率，第二部分的计算可以交由决策中心执行。这样，LODS 的决策时间虽然远高于一致性哈希，但对于用户请求的响应时间只有极小程度的降低，从而有效地保证了对于用户请求处理的时效性。

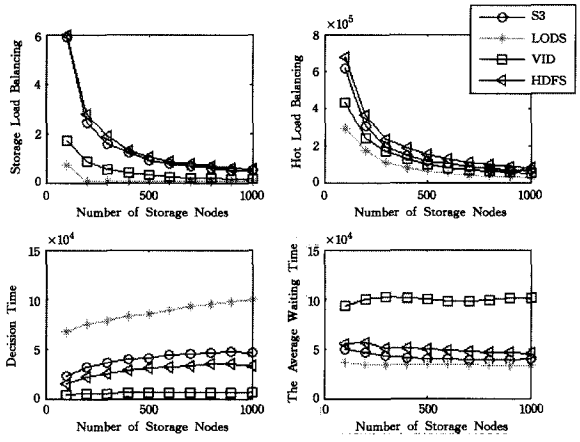

图 11 4 种副本分布策略存储负载平衡性、热度负载平衡性、决策时间、平均等待时间的性能对比

运用副本分布策略性能评价的综合指标，并假设存储负载平衡、热度负载平衡、决策时间和副本存储请求的平均等待

时间四者重要性的成对比较矩阵为：

$$S=\begin{pmatrix} 1 & 1 & 2 & 1 \\ 1 & 1 & 2 & 1 \\ 0.5 & 0.5 & 1 & 0.5 \\ 1 & 1 & 2 & 1 \end{pmatrix}$$

通过实验，可以得到随存储节点数量的增加 Amazon S3、HDFS、VID、LODS 四者的组合权重的变化趋势，如图 12 所示。在成对比较矩阵为 S 的条件下，LODS 的组合权重低于其余 3 种副本分布策略，说明副本分布策略的综合性评价方法可以很好地对副本分布策略进行综合评价。

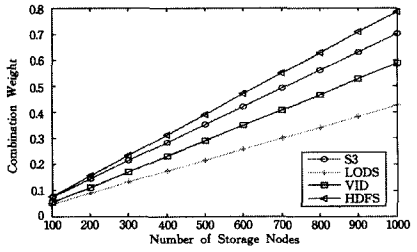

图 12　4 种副本分布策略综合性能对比

结束语　本文对云存储中副本分布进行研究，提出了一种基于多目标优化的副本分布 LODS 策略。本策略将一致性哈希法与层次分析法有效地结合在一起，一方面利用一致性哈希使副本分布在总体上相对均匀，另一方面利用层次分析法对副本分布的多个目标进行优化，从而使其在局部多个目标的综合性能方面达到最优。进一步，本文提出了一套基于层次分析法的副本分布策略综合评价指标，克服了传统方法只能对单一目标进行评价的不足。仿真实验说明，LODS 提高了云存储系统存储负载平衡性、热度负载平衡性和平均等待时间 3 项性能，在综合性能上更具有优势。

参 考 文 献

[1] Borthakur D. The Hadoop distributed file system: Architecture and design [EB/OL]. [2013-08-04]. http://hadoop.apache.org/docs/r1.2.1/hdfs_design.pdf

[2] Ghemawat S, Gogioff H, Leung P T. The google file system [C]// Proceedings of the 19th ACM Symp on Operating Systems Principles. New York: ACM, 2003: 29-43

[3] Amazon. comInc. Amazon simple storage service (AmazonS3) [EB/OL]. [2014-04-09]. http://aws.amazon.com/s3

[4] Lewn D. Consistent hashing and random trees: Algorithms for caching in distributed networks[D]. Cambridge, Massachusetts: Massa-chusetts Institute of Technology, Department of Electrical Engineering and Computer science, 1998

[5] 郭金玉，张忠彬，孙庆云. 层次分析法的研究与应用[J]. 中国安全科学学报，2008，18(5)：148-153

[6] Mohamed N, Al-Jaroodi J, Eid A. A dual-direction technique for fast file downloads with dynamic load balancing in the Cloud [J]. Journal of Network and Computer Applications, 2013, 36(4): 1116-1130

[7] 周敬利，周正达. 改进的云存储系统数据分布策略[J]. 计算机应用，2012，32(2)：309-312

[8] 王永洲，茅苏. HDFS 中的一种数据放置策略[J]. 计算机技术与发展，2013，(5)：90-92

[9] 董继光，陈卫卫，田浪军，等. 大规模云存储系统副本布局研究[J]. 计算机应用，2012，32(3)：620-624

[10] 董继光，陈卫卫，吴海佳，等. 基于动态副本技术的云存储负载均衡研究[J]. 计算机应用研究，2012，29(9)：3422-3424，3436

[11] Chen Tao, Bahsoon R, Tawil A-R H. Scalable service-oriented replication with flexible consistency guarantee in the cloud[J]. Information Sciences, 2014, 264: 349-370

[12] Du Zhi-hui, Hu Jing-kun, Chen Yi-nong, et al. Optimized QoS-aware replica placement heuristics and applications in astronomy data grid [J]. Journal of Systems and Software, 2011, 84(7): 1224-1232

[13] 廖彬，于炯，张陶，等. 基于分布式文件系统 HDFS 的节能算法[J]. 计算机学报，2013，36(5)：1047-1064

[14] Gkantsidis C, Vytiniotis D, Hodson O, et al. Rhea: automatic filtering for unstructured cloud storage[C]// Presented as part of the 10th USENIX Symposium on Networked Systems Design and Implementation. 2013: 343-355

[15] Freedman M J, Shaikh A. Performance isolation and fairness for multi-tenant cloud storage[C]// Proc. 10th USENIX Conference on Operating Systems Design and Implementation. 2012: 349-362

[16] 高尚. 3 种计算层次分析法中权值的方法[J]. 科学技术与工程，2007，7(20)：5204-5207

[17] Dong Bo, Zheng Qing-hua, Tian Feng, et al. An optimized approach for storing and accessing small files on cloud storage[J]. Journal of Network and Computer Applications, 2012, 35(6): 1847-1862

(上接第 43 页)

[5] 王玥，蔡皖东，段琪. 一种自适应动态负载均衡算法[J]. 计算机工程与应用，2006，11(21)：121-123

[6] 许少华，夏智伟. 基于轮转周期的动态反馈负载均衡算法[J]. 计算机技术与发展，2013，23(6)：55-59

[7] 梁彪，黄战. 基于实时性能动态反馈的负载均衡算法[J]. 计算机系统应用，2010，19(3)：183-186

[8] 肖海良. 基于用户体验的移动终端整机性能评测研究[J]. 移动通信，2012，13(3)：71-74

[9] 汤克明，王创伟. P2P 模拟器的比较研究[J]. 微电子学与计算机，2008，25(9)：105-108

[10] Zhang Yong-bing, Das S K. An efficient load- balancing algorithm based on a two-threshold cell selection scheme in mobile cellular networks[J]. Computer Communications, 2000, 23(5): 452-461

[11] Chakrabarti G, Kulkarni S. Load balancing and resource reservation in mobile and hoc networks[J]. Ad hoc Networks, 2006, 4(2): 186-203

[12] Askarian C, Beigy H. ASurvey For Load Balancing in Mobile WiMAX Networks[J]. Advanced Computing: An International Journal(ACIJ), 2012, 3(2): 119-137

[13] Kacimi R, Dhaou R. Load balancing techniques for lifetime maximizing in wireless sensor networks[J]. Ad hoc Networks, 2013, 11(8): 2172-2186

[14] Toh C K, Le A N, Cho Y Z. Load Balanced Routing Protocol for Ad Hoc Mobile Wireless Networks [J]. IEEE Communications Magazine, 2009, 47(8): 78-84