云存储系统中最小开销的数据副本布局转换策略

吴修国 刘 翠
(山东财经大学管理科学与工程学院 济南 250014)

摘 要 副本技术是提高云存储系统中数据可靠性访问和系统容错性的常用策略。依据用户需求以及环境变化,及时对数据副本布局进行动态调整,是目前副本管理研究的重要内容之一。然而,现有研究大都以副本布局转换是自动完成的为前提,仅关注于数据副本数目与位置等副本布局方案设计,较少涉及副本布局转换的任务调度问题。事实上,副本布局转换是有关多数据中心数据副本迁移与删除操作的复杂任务调度问题,不同的任务调度策略占用的空间、时间不同,由此导致成本、效率等存在较大差异。基于此,首先给出云存储系统中面向多数据中心的数据副本布局转换任务调度模型,以及该问题的可行性分析。然后,从降低成本的角度给出最小开销的数据副本布局转换任务调度问题的定义,并基于 0-1 背包问题证明其是 NP 完全的。在此基础上,给出随机(Random)、最小传输开销优先(MTCF)、最大机会成本优先(MOCF)以及同数据最小传输成本优先(MTCFSD)等副本布局转换任务调度策略。最后,以 CloudSim 为仿真平台进行了模拟实验,结果表明,最小开销的数据副本布局转换策略与同类算法相比,在传输次数上减少了约 60%,相对开销降低了约 50%,证明了转换策略的可靠性与有效性,从而进一步提升了云存储系统的性能。

关键词 云存储系统,副本,布局转换,任务调度,成本
中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/jsjkx.180901623

Data Replicas Distribution Transition Strategy in Cloud Storage System

WU Xiu-guo LIU Cui
(School of Management Science and Engineering, Shandong University of Finance and Economics, Jinan 250014, China)

Abstract Replication is a common method used to improve the data access reliability and system fault tolerance in the cloud storage system. It is one of the most important topics to reschedule the replicas distribution dynamically according to the changes of users' requirements and environment in the replicas management. However, current replicas redistribution strategies mostly focus on the new replicas schemes, such as replicas number and their placements, on the premise that it can be completed automatically, without taking into account the task scheduling problem in practice. In fact, data replica distribution transition is a complex scheduling problem involving data replicas migration and deletion among data centers. In addition, the required disk space and time of different scheduling strategies have large differences, leading to big difference in cost and efficiency. In this way, this paper first proposed a data replicas distribution transition model and feasibility analysis in the cloud storage environment. Also a minimum-cost data replicas distribution transition problem definition was presented, and then its complexity was proven based on 0-1 Knapsack problem. Besides random strategy, three transition strategies (MTCF, MOCF and MTCFSD) were given from minimum-cost view. In the end, a series of experiments were performed on CloudSim simulation platform. The results show that nearly 60% of transmission number and 50% of transmission cost are reduced compared with other methods, indicating the proposed method's reliability and effectiveness, so as to further improve the cloud storage system performance.

Keywords Cloud storage system, Replicas, Distribution transition, Tasks scheduling, Cost

1 引言

为提高云存储系统的可靠性、可用性以及扩展性,通常采用对数据设置副本(Data Replicas)的方式在不同地理位置(数据中心)为共享资源提供多个数据映像,以减少数据传输时间和带宽消耗,提高数据资源的访问可靠性和系统容错性[1]。然而,在动态分布式云存储系统中,数据访问具有随机性和突发性,数据副本的热度会随之发生变化;与此同时,数据中心以及网络带宽等失效事件的发生同样具有不确定性,从而使得数据副本布局调整成为数据副本管理的重要内容之

到稿日期:2018-09-03 返修日期:2019-01-29 本文受国家自然科学基金(61571272),山东省自然基金(ZR2016FM01)资助。
吴修国(1975—),男,博士,副教授,主要研究方向为云计算环境下的数据管理以及任务调度等,E-mail:xiuguosd@163.com(通信作者);**刘 翠**(1994—),女,硕士生,主要研究方向为云环境下的任务调度等。

一[2]。为此，许多研究者提出了不同的数据副本调整方案，即新的数据副本位置以及选择策略[3-5]，并取得了较好的效果。尽管如此，当前的研究通常假定数据中心的资源是无限的，数据副本布局的调整是“即刻”完成的，不需要时间成本与空间成本，对如何从原有副本布局转换为新的布局方案（本文称为副本布局转换），即调整的具体步骤、调整产生的成本以及调整后的完整性验证等均没有涉及。随着网络规模的扩大以及数据量的增加，数据副本布局转换需要进行大量的数据副本迁移与删除操作，而且迁移与删除等操作之间还存在时序上的约束。同时，在数据副本布局的转换过程中，不可避免地要占用云存储系统的资源，不同的转换策略在存储、传输等成本上存在着巨大的差异。由此，副本布局转换是一个有关成本、空间、时间等多方面因素的离散约束优化调度问题。研究一种面向多数据副本的副本布局任务调度策略，对提高云存储系统的性能具有重要意义。

基于此，在分析云存储系统中数据副本布局转换过程及其存在问题的基础上，提出面向多数据中心的数据副本布局转换任务调度模型，并从降低成本的角度给出了最小成本的任务调度问题的定义，随后，提出近似最小成本的副本布局转换任务调度算法。

本文第 2 节对相关研究进行综述；第 3 节给出云环境下数据副本布局转换模型以及最小成本布局转换问题的定义，并对其复杂性进行分析；第 4 节提出云环境下数据副本布局转换任务调度算法；第 5 节给出实验与模拟；最后总结全文并对下一步工作进行阐述。

2　相关研究

在动态多变的云计算环境下，数据副本布局调整是数据副本管理的重要内容。目前研究者多以网格[3]、CDN[6]等为应用背景，以预先设定的响应时间阈值[7]、访问频度阈值[8]、网络负载变化需求[9-10]等性能指标作为副本调整的依据，提出新的数据副本布局方案。这些方案对响应环境变化、缩短访问时间、提高数据可用性等具有积极意义，但并未给出具体的实施方案。根据前期分析，数据副本删除与传输是数据副本布局的主要内容。数据副本的删除相对简单，在确保不会作为数据源传输时，由各数据中心实施即可。然而，数据副本的传输涉及到传输的顺序与路径等，是一个复杂调度问题。传统的调度算法（如 Min-min[11] 与 Max-min[12]）、基于 DAG 的 MapReduce 调度算法[13]等在一定程度上提高了系统调度效率，然而，这些方法难以直接应用在云存储系统数据副本布局转换的任务调度中，其原因在于：1）副本转换中的数据传输任务涉及到两个甚至两个以上的数据中心（主机），而传统的任务调度是把任务分配到某一台主机上；2）任务调度包含传输与删除两种类型的任务，某些数据可能会被看作是多个副本的迁移来源，因此具有严格的时序关系；3）不同的副本布局转换策略占用的空间、时间不同，在确定转换策略时要兼顾成本、可行性、效率等要素，以更好地满足云存储系统的要求；4）转换中的负载均衡显得更加重要，因为迁移的时机也会受到数据中心“忙”的约束，最佳的方案是在空闲时实施副本迁移与删除操作。

图论是求解多任务调度问题的有效工具。文献[14]利用边着色方法给出了一种分布式环境下近似最优的数据传输调度算法，该算法可有效解决数据传输的效率问题；在此基础上，文献[15]利用启发式算法给出了一种近似最优的数据传输策略；文献[16]则通过构建最大斯坦纳树（Steiner tree），给出了数据传输路径最短的传输策略。然而，上述研究均假设所有的数据大小固定，并且网络结构是一个完全图，不具有通用性。此外，建立整数规划模型，并以启发式算法求近似最优解，也是解决这类问题的方法之一。文献[17]对该问题建立了混合整数规划模型，并给出了启发式调度算法，然而，研究对数据大小与网络结构进行了限定，同时也没有考虑副本布局调整时占用的空间以及产生的成本等问题。当前专门对数据副本布局转换进行研究的文献非常少，文献[19]是其中一篇，然而，其仅对副本布局转换的可行性进行了分析，并没有提出具体的实施步骤，也没有考虑用户请求的相关应用特性、成本以及调度效率等。

综上，在确定数据副本布局转换策略时，由于网络规模的扩大以及数据量的增加，需要进行大量的数据迁移与删除操作，而且这些操作之间还存在时序上的约束；同时，在数据副本布局调整过程中，不可避免地会占用云存储系统中的数据中心资源，有的数据副本可能会被删除，有的数据副本可能需要新建。现有文献还没有对数据副本布局转换进行研究，特别是从数据成本的角度提出副本布局的转换策略。基于此，本文给出面向多数据中心的副本布局转换任务调度模型，并从降低成本的角度给出了最小成本的副本布局转换策略，以提升云存储系统的性能。

3　云存储系统中数据副本布局转换问题的建模

本节对云存储系统中的数据副本布局转换问题进行了建模；之后，分别对最小成本的数据副本布局转换问题的可行性和复杂性进行了分析。

3.1　副本布局转换任务调度问题的建模

3.1.1　RDTSP 模型

云存储系统是由多个分布在不同地理位置的数据中心（Data Center）dc 构成，各数据中心包含了规模庞大的计算资源和存储设施为用户提供相应的计算和存储服务，从而实现信息的集中处理、存储、交换、传输及管理等功能。数据（Data）d 及其副本（Replicas）r 存储在数据中心上。图 1 给出了由 3 个数据中心 dc_1，dc_2 和 dc_3 构成的云存储系统中多数据副本的数据访问模型，可以看出，用户调用数据 d_2 时，先要向最近的数据中心 dc_1 发出请求，如果该数据中心存有副本，则直接返回；否则，继续寻找最近的数据中心 dc_2，直到找到数据 d_2 为止。

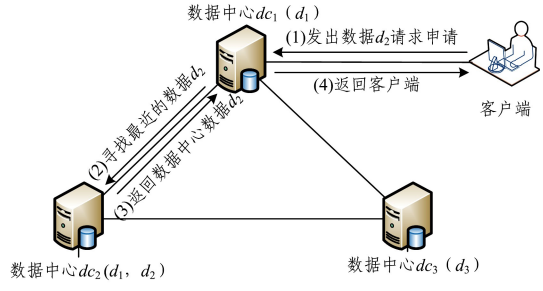


图 1　云环境下多数据副本的数据访问模型

Fig. 1　Data access model of multi-replicas in cloud

假设云存储环境中有 M 个数据中心 $dc_i(i=1,2,\cdots,M)$，其存储空间为 $s(dc_i)$；N 个数据 $d_k(k=1,2,\cdots,N)$ 的大小为 $s(d_k)$；二维矩阵 $\boldsymbol{X}_{M\times N}$ 用于描述当前数据副本的布局 X，定义如下：

$$X=[x_{ik}]=\begin{cases}1, & dc_i \text{ 上放置数据 } d_k \text{ 的副本}\\ 0, & \text{否则}\end{cases}$$

用 $S_{N(i,k,X)}$ 和 $S_{N2(i,k,X)}$ 分别表示当前布局 X 下距离数据中心 dc_i 最近和次近的存放数据 d_k 副本的数据中心标识。当然，这里的最近并不一定是距离最短，它是根据系统需求定义的某方面的性能，如时间最短、成本最小等。

经过分析可知，可以把数据副本布局转换的任务分为两种类型。

1)数据副本传输任务 T_{ikj}：将数据 d_k 由数据中心 dc_j 传输至数据中心 dc_i；

2)数据副本删除任务 D_{ik}：将数据 d_k 从数据中心 dc_i 中删除。

由此，数据副本布局转换任务调度可以看作是由数据副本传输与删除任务构成的调度集合 $H=\{A_1\rightarrow A_2\rightarrow A_3,\cdots,\rightarrow A_m\}$，其中动作 $A_i\in\{T_{ikj},D_{ik}\}$，T_{ikj} 和 D_{ik} 分别代表数据副本传输和删除任务。

进一步地，用 X^u 和 X^{u+1} 分别表示转换前后的副本布局情况。显然，一个传输任务 $A_u=T_{ikj}$ 是有效的，当且仅当满足以下条件：

1)在副本布局 X^u 中，dc_j 存放了数据副本 d_k，即 $X^u_{jk}=1$；

2)在副本布局 X^u 中，dc_i 没有存放数据副本 d_k，即 $X^u_{ik}=0$；

3)在副本布局 X^{u+1} 中，dc_i 存放数据副本 d_k，即 $X^{u+1}_{ik}=1$；

4)数据中心 dc_i 有充足的存储空间，即 $s(dc_i)-\sum_{\forall k'}X^u_{ik'}s(d_{k'})>=s(d_k)$。

同样地，一个删除任务 $A_u=D_{ik}$ 是有效的，当且仅当满足以下条件：

1)在副本布局 X^u 中，数据中心 dc_i 存放了数据副本 d_k，即 $X^u_{ik}=1$；

2)在副本布局 X^{u+1} 中，数据中心 dc_i 不需存放数据副本 d_k，即 $X^{u+1}_{ik}=0$。

每一个任务调度完成后，都将使副本布局由 X^u 向 X^{u+1} 产生改变，即 $A_u=T_{ikj}\Rightarrow X^{u+1}_{ik}=1$，$A_u=D_{ik}\Rightarrow X^{u+1}_{ik}=0$。一个任务调度 $H=\{A_1,A_2,\cdots,A_t\}$ 是由副本布局 X^{old} 转换到 X^{new} 的有效任务调度，当且仅当其中的每一个任务都是有效的。

定义 1(副本布局转换任务调度问题(Replicas Distribution Transition Scheduling Problem，RDTSP)) 云计算环境下，给定数据副本布局 X^{old} 和 X^{new}，确定由数据传输任务与删除任务组成的有效任务调度序列 H，使任务执行后数据副本布局由 X^{old} 转换为 X^{new}，称为副本布局转换任务调度问题。

3.1.2 RDTSP 问题可行性分析

一般来说，数据 d_k 存放在云存储环境中都有一个主数据中心(Primary Data Center)，记为 pdc_k，由它负责管理数据 d_k 的副本布局以及更新等，并约定该数据作为原始数据不会被删除。最简单的调度就是先删除各数据中心上所有多余的数据副本；然后依次从主数据中心将数据副本传输到各需要放置数据的数据中心。然而，当不存在主数据中心，并且数据中心的存储空间有限时，副本布局的转换并不一定总是有解。考查图 2 所示的副本布局转换的示例。

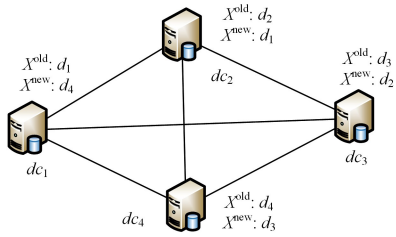


图 2 RDTSP 问题的示例

Fig. 2 Example of RDTSP

图 2 包含 4 个数据中心 dc_1，dc_2，dc_3 和 dc_4，以及 4 个大小相同的数据对象 d_1，d_2，d_3 和 d_4。由于存储空间有限，每个数据中心只能放置一个数据对象。可以得知，该布局转换问题没有可行解，即不存在有效的任务调度满足由副本布局 X^{old} 到 X^{new} 的转换。为更清楚地说明，将每个数据中心映射为一个节点，将每一个数据传输任务映射为一条有向边，可以将其转换为一个任务调度图，如图 3 所示。

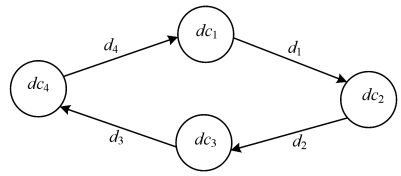


图 3 RDTSP 问题的可行性分析

Fig. 3 Feasibility analysis of RDTSP

显然，从图 3 可以看出，实施数据副本布局转换的有效任务调度队列是不存在的，即为了把数据 d_4 放置到 dc_1 中，需要预先把 d_1 传输到 dc_2；而要把 d_1 放置到 dc_2 中，则需要先把 d_2 传输到 dc_3；依此类推，形成了死锁，无法找到有效解。为此，数据副本布局转换是否存在有效调度，需要考虑数据大小、数据中心存储空间、数据传输路径选择等多个要素。

3.2 最小成本的副本布局转换任务调度问题的建模

3.2.1 MRDTSP 模型

云计算采用按需服务的计费模式(pay-as-you-go)，成本是决定云计算环境下数据存储服务质量的重要要素之一。数据副本转换过程包含数据的传输、删除以及完整性验证等，不可避免地会产生一定的成本。一般来说，数据副本转换过程产生的成本主要包括以下几方面。

1)读取成本，或称为传输开销，指当用户发出数据调用请求后，数据由数据中心传输到用户链接数据中心的开销。数据调用时采用点对点的方式传输，假设 $l(dc_i,dc_j)$ 为单位数据在数据中心 dc_i 和 dc_j 之间的传输开销，并约定 l_{ij} 的大小不变，且有 $l_{ij}=l_{ji}$。

2)删除开销，指在数据副本布局转换过程中，将多余数据副本从数据中心删除时耗费的成本。

3)验证开销，指在数据副本布局转换完成后，由主数据中心 pdc_k 对各副本所在数据中心进行校验的成本。

数据 d_k 的一次读取成本可以描述为：

$$C_r^i=(1-X_{ik}^{\text{old}})\times X_{ik}^{\text{new}}\times l(i,S_{N(i,k,X)})\times s(d_k)$$

其中，$l(i,S_{N(i,k,X)})$为数据中心 dc_i 与距其最近且存储了数据 d_k 副本的数据中心 $S_{N(i,k,x)}$ 之间的单位数据大小的传输成本。而删除成本为：

$$C^d=X_{ik}^{\text{old}}\times(1-X_{ik}^{\text{new}})\times\mu\times s(d_k)$$

验证成本主要是传输少量数据对数据副本进行校验产生的成本，可忽略不计。

由此，假设 $C(H_u)$表示调度序列中操作 u^{th} 的开销，则数据副本转换任务调度序列 $H=\{A,A_2,\cdots,A_t\}$ 的总开销为：

$$C=\sum_{u=1}^{l}(C(H_u)) \tag{1}$$

定义 2（最小成本的副本布局转换任务调度问题（Minimum-cost Replicas Distribution Transition Scheduling Problem，MRDTSP））　云计算环境下，给定数据副本布局 X^{old} 和 X^{new}，确定由数据传输任务与删除任务组成的有效任务调度序列 H，并使得该任务的调度开销值最小，即式(1)的值最小，称其为最小成本的副本布局转换任务调度问题。

3.2.2　MRDTSP 问题的复杂性分析

很显然，MRDTSP 问题是一个 NP 问题。进一步地，可以将经典(0，1)背包问题（Knapsack Problem，KP）归约为 MRDTSP 问题，从而证明该问题是 NP 完全的。

对于每一个背包问题的实例，都可以通过构造一个云存储网络结构来使背包问题的解同样也是 MRDTSP 问题的解。具体构造方法包含以下 5 个步骤。

1）构建一个由 $n+3$ 个数据中心组成的云存储网络，数据中心标识为 $dc_1,dc_2,\cdots,dc_n,dc_{n+1},dc_{n+2},dc_{n+3}$。

2）该云存储网络中包含 $n+1$ 个数据对象，将背包问题中的 n 个物品映射为 MRDTSP 问题中的 n 个数据 $d_1,d_2,\cdots,d_n$，并要求 $s(d_i)=s_i$，数据对象 d_{n+1} 是 $d_1,d_2,\cdots,d_n$ 的大小之和，即 $s(d_{n+1})=\sum_{1\leqslant i\leqslant n}s(d_i)$。

3）在数据副本布局 X^{old} 下，数据对象 $d_1,d_2,\cdots,d_n$ 分别存储在数据中心 $dc_1,dc_2,\cdots,dc_n$；数据中心 dc_{n+1},dc_{n+2} 以及 dc_{n+3} 的空间大小以及数据副本放置如表 1 所列。其中，S 为背包问题给出的体积。

表 1　数据中心的大小以及数据放置

Table 1　Data center size and data placement

数据中心	空间大小	数据放置
dc_{n+1}	$S+\sum_{1\leqslant i\leqslant n}s(d_i)$	d_{n+1}
dc_{n+2}	$\sum_{1\leqslant i\leqslant n}s(d_i)$	$d_1,d_2,\cdots,d_n$
dc_{n+3}	$\sum_{1\leqslant i\leqslant n}s(d_i)$	d_{n+1}

4）数据中心 dc_{n+1} 和 dc_{n+2} 之间的单位数据传输开销为 1；数据中心 $dc_1,dc_2,\cdots,dc_n$ 到数据中心 dc_{n+1} 的单位数据传输开销为 $l_{in+1}=b_i'$，$1\leqslant i\leqslant n$，$b_i'=\dfrac{b_i\prod_{1\leqslant i\leqslant n}s(d_i)}{s(d_i)}$；数据中心 d_{n+2} 和 d_{n+2} 之间的单位数据传输开销为 $\sum_{1\leqslant i\leqslant n}(b_i'+1)$，如图 4 所示。

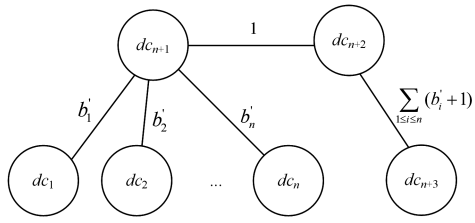


图 4　背包问题归约为 MRDTSP 问题构建的云存储结构图

Fig. 4　Cloud storage structure for reducing Knapsack to MRDTSP

5）副本布局 X^{new} 是在原布局 X^{old} 的基础上，将数据中心 dc_{n+1} 和 dc_{n+2} 中的数据对换，即在 dc_{n+1} 中放置背包问题的所有数据，在 dc_{n+2} 中放置数据 d_{n+1}。

假设最小成本的副本布局转换任务调度为 H-OPT，其他副本布局转换任务调度为 H'，$H'=\{A_1,A_2,\cdots\}$，其中 $A_1=T_{(n+2)(n+1)(n+1)}$，即把数据 d_{n+1} 由数据中心 dc_{n+1} 传到数据中心 dc_{n+2}；$A_2=T_{(n+2)(n+1)(i)}$，…。可以看出，任务 $T_{(n+2)(n+1)(n+3)}\notin$ H-OPT，其原因在于：该任务 $T_{(n+2)(n+1)(n+3)}$ 的开销为 $l_{(n+2)(n+1)(n+3)}=\sum_{1\leqslant i\leqslant n}(b_i'+1)\times\sum_{1\leqslant i\leqslant n}s(d_i)$；任务 $T_{(n+2)(n+1)(n+1)}$ 以及 $T_{(n+1)(i)(i)}$ 的开销为：$l_{(n+2)(n+1)(n+1)}+\sum_{1\leqslant i\leqslant k}b_k's(d_k)$，而前者比后者大，因此 $T_{(n+2)(n+1)(n+1)}\in$ H-OPT。

另一方面，$H'\neq$H-OPT，这是因为数据中心 dc_{n+1} 的空余存储空间为 S，至少能够预先从 dc_{n+2} 传输一个背包问题数据 d_i 到数据中心 dc_{n+1}，而其开销 $s(d_i)\times 1$ 要小于 H' 中从 dc_i 传输到 dc_{n+1} 的开销。因此，H-OPT 应该是先把部分背包问题数据从 dc_{n+2} 传输到 dc_{n+1}，然后把数据 d_{n+1} 从数据中心 dc_{n+1} 传输到 dc_{n+2}，最后把剩余的数据由 dc_i 传输到 dc_{n+1}。用 I-OPT 表示 H-OPT 的总开销，则有：

$$\begin{aligned}\text{I-OPT}(X^{\text{old}},X^{\text{new}})&=\sum_{i\in W'\wedge i\neq n+1}s(d_i)+s(d_{n+1})+\sum_{i\notin W'\wedge i\neq n+1}b_i's(d_i)\\&=\sum_{i\in W'\wedge i\neq n+1}s(d_i)+\sum_{1\leqslant i\leqslant n}s(d_i)+\sum_{i\in W'\wedge i\neq n+1}b_i's(d_i)\end{aligned} \tag{2}$$

然而，要保证 H-OPT 的转换开销最小，就要求 W' 尽量大，从而使得式(2)的值尽量小。利用前面的定义替换掉 b_i'，可以将式(2)转换为：

$$\text{Min}\Big(\sum_{i\in W'\wedge i\neq n+1}s(d_i)+\sum_{1\leqslant i\leqslant n}s(d_i)+\prod_{1\leqslant i\leqslant n}s(d_i)\sum_{i\in W'\wedge i\neq n+1}b_i\Big) \tag{3}$$

由于 $\sum_{1\leqslant i\leqslant n}s(d_i)$ 为常量，因此要求 $\sum_{i\in W'\wedge i\neq n+1}s(d)+\prod_{1\leqslant i\leqslant n}s(d_i)\sum_{i\in W'\wedge i\neq n+1}b_i$ 最小。

同时，由于 $\prod_{1\leqslant i\leqslant n}s(d_i)b_i\geqslant s(d_i)$，$1\leqslant i\leqslant n$，以及 $\sum_{1\leqslant i\leqslant n}s(d_i)$ 为常量，因此式(3)可以简化为 $\text{Min}(\sum_{i\in W'\wedge i\neq n+1}b_i)$。由此，H-OPT 的最小化成本问题的可以等价转换为最大化 $\sum_{i\in W\wedge i\neq n+1}b_i$，即背包问题的优化问题。

综上，给定一个(0，1)背包问题实例，可以在多项式时间内依据上述方法构建一个云存储网络实例，并可以回答：是否存在一个副本布局转换任务调度 H 满足 $\text{I-OPT}(X^{\text{old}},X^{\text{new}})\leqslant\sum_{\forall i}s_i+\sum_{\forall i}(b_i-K)\prod_{\forall i}s_i+S$，如果存在一个调度 H，则一定存在背包问题的解。由此可得，定理 1。

定理 1 MRDTSP 问题是 NP 完全问题。

设当前数据副本布局 X^{old}，更新后的布局 X^{new}；C_{sum}^{old} 和 C_{sum}^{new} 分别表示两种布局下的数据副本管理总成本，那么收益函数可以定义为：$B(old,new)=C_{sum}^{old}-C_{sum}^{new}-I(old,new)$，其中，$I(old,new)$表示布局转换的成本。显然，$B(old,new)>0$ 是副本布局调整的必要条件之一，即当前的总管理成本大于转换后的管理成本与转换所需的成本之和。

4 云存储系统中数据副本布局转换任务调度算法

本节提出一种近似最优的副本布局转换任务调度策略。该策略首先构建初始的任务调度队列，然后对该初始队列进行优化。

4.1 构建初始任务调度队列

构建初始任务调度队列的目标是在不考虑任务调度开销的前提下，预先生成一个有效的任务调度。初始任务调度队列算法以数据副本布局 X^{old} 以及 X^{new} 为输入，以有关数据副本传输任务 T_{ikj} 和删除任务 D_{ik} 的调度队列为输出。初始任务队列构建算法的描述如算法 1 所示。

算法 1 初始任务队列构建算法(Initial Schedule Building Algorithm，ISBA)

输入：数据副本布局 X^{old}，X^{new}

输出：初始任务调度 H

1. H←Ø；
2. X←X^{old}；
3. While (Exists(T_{ikj}))
4. {{选取待添加任务 T_{ikj}；
5. If($s(d_k)+\sum_{1\leqslant i\leqslant M}s(d_i)<s(dc_i)$)
6. {H←H∪T_{ikj}；
7. $X_{ik}=1$；}
8. Else
9. {选取待删除任务 D_{ik}；
10. H←H∪D_{ik}；
11. $X_{ik}=1$；}}
12. While (Exists(D_{ik}))
13. {选取剩余待删除任务 D_{ik}；
14. H←H∪D_{ik}；
15. $X_{ik}=1$；}
16. End.

显然，算法 1 是一种贪婪算法。一般来说，有以下 4 种策略。

1)任务随机选取(Random)策略。该策略在算法 1 的第 4 行和第 9 行，不论是待添加任务的选取还是待删除任务的选取都是随机的。

2)最小传输开销优先(Minimum Transfer Cost First，MTCF)策略。在算法 1 的第 9 行进行数据副本删除时，有些待删除数据副本还可能作为某些数据传输的数据源，如果过早删除它们，从其他数据源进行数据传输的开销会增大。该策略在选取传输任务时，选取作为候选数据源时开销最小的数据副本，其目的是一旦删除这些数据，在将来的数据传输中将尽可能地降低额外开销。为此，为每一个数据中心 dc_i 中的待删除副本 d_k 定义开销值 B_{ik}，其值为删除掉该数据后从其他数据源进行传输的开销和减去该数据作为数据源传输到其他数据中心的开销和：

$$Bik=s(d_k)\times\left(\sum_{\forall j,N(j,k,X)}l_{jN2(j,k,X)}-l_{jN(j,k,X)}\right) \tag{4}$$

算法执行过程中，每次总是选择最小开销的数据中心 dc_i 进行处理，而在删除数据时也总是按照开销由低到高的顺序进行。

3)最大机会成本优先(Maximum Opportunity Cost First，MOCF)策略。在算法 1 的第 4 行选择待增加副本任务时，优先选择以最有可能被删除的数据副本作为数据源的任务，因为其一旦删除，该任务需要从次近副本传输，增加了额外开销。因此，为每一个需要添加数据副本的数据中心 dc_i 定义数据 d_k 的机会成本为 V_{ik}：如果 $dc_{N(i,k,X)}$ 不是一个待删除数据副本，则 $V_{ik}=0$；否则，它的大小是从次近的数据源传输到 dc_i 与最近数据源传输到 dc_i 的差值，即 $V_{ik}=s(d_k)(l_{iN2(i,k,X)}-l_{iN(i,k,X)})$。在每一次循环中，待增加副本的选择总是选择机会成本最高的数据。如果由于存储空间原因需提前删除一些待删除副本，那么副本的选择可以按照式(4)中的顺序进行。

4)同数据最小传输成本优先(Minimum Transfer Cost First with Same Data，MTCFSD)策略。该策略首先随机选择一个数据 d_k，之后将数据传输到所有需要添加副本的数据中心上。在每一次循环中，总是选择传输开销最小的数据中心作为数据目标，一旦发现存储空间不足，将删除最小传输成本的数据副本。

4.2 任务调度队列的优化

在构建初始任务调度队列的基础上，对其进行进一步的优化，在保证有效性的前提下尽可能降低任务开销。任务调度队列优化算法以副本的初始布局 X^{old}、新布局 X^{new} 以及初始调度队列 H^{old} 作为输入，以优化后的调度队列 H^{new} 作为输出。任务调度队列优化算法的描述如算法 2 所示。

算法 2 任务调度队列优化算法(Improving Schedule Algorithm，ISA)

输入：数据副本布局 X^{old} 与 X^{new}，初始任务调度队列 H^{old}

输出：优化后的任务调度队列 H^{new}

1. t′←Ø；
2. t←T_{ijk}；
3. While (!H^{old}.End)
4. {t′←$T_{i'kj'}$；
5. If ($l_{i'i}>l_{i'j'}$)
6. H^{new}=Switch(t′，t)；
7. ValidCheck(H^{new})；
8. }
9. End.

算法 2 从调度队列 H_1^{old} 的第一个任务 $T_{i'kj'}$ 开始，从左至右进行扫描，直到找到同样为待添加数据 d_k 的任务 T_{ikj} 为止。假定 $H=\{\cdots,T_{i'kj'},G_1,T_{ikj},G_2,\cdots\}$，其中 G_1 和 G_2 为子调度集。为降低调度开销，首先考查待传输任务 $T_{i'kj'}$ 和 T_{ikj} 的顺序能否调换，即先执行任务 T_{ikj}；然后执行任务 $T_{i'kj'}$。对每一个任务 $T_{i''kj''}\in\{T_{i'kj'},G_1,G_2\}$，定义待传输任务收益 $b:=$

0；当且仅当 $l_{i'i}>l_{i'j'}$；否则，其值为由当前的数据中心 dc_i 或者 dc_j 传输到 $dc_{i'}$ 的开销之差，即 $s(d_k)\times(l_{i'j'}-l_{i'i})$，通过将待传输任务 T_{ikj} 提前，来降低后期任务 $T_{i'kj'}$ 的开销，如图 5 所示。

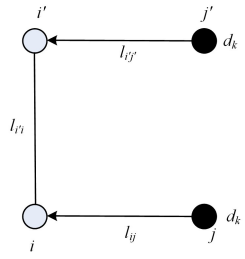


图 5　调度队列优化示意

Fig. 5　Example of scheduling queue optimization

5　仿真实验与分析

本节首先给出仿真实验环境的配置，然后进行仿真实验并分析结果。

5.1　实验环境

为测试云存储系统中数据副本布局任务转换策略的有效性，采用 CloudSim 3.0 模拟一个云存储系统的局部域，进行仿真实验[19]。虚拟机数据中心数为 20，可用数据个数为 100，大小为 10～20 GB。数据副本布局转换前后 X^{old} 与 X^{new} 的参数如表 2 所列。

表 2　数据副本布局 X^{old} 与 X^{new} 的参数

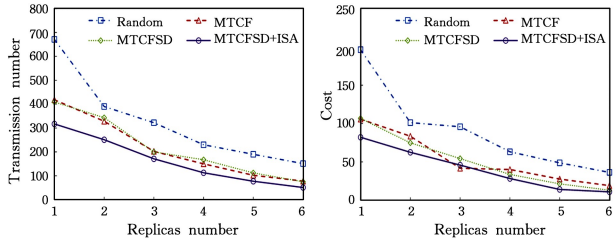
Table 2　Parameters of data replication distribution X^{old} and X^{new}

参数名称	参数范围
数据数目	[50，100]
副本数目	[1，6]
初始位置	随机
目标位置	随机

实验共进行了 3 组，主要从数据传输次数（N_t）与布局转换开销（C_s）两个方面对不同的副本布局转换策略进行比较，包括随机（Random）、最小传输开销优先（MTCF）、同数据最小传输开销优先（MTCFSD）及其优化（MTCFSD＋ISA）等副本布局转换策略等[19-20]。数据传输次数是指数据副本布局调整过程中所有的数据副本传输次数之和，包括中间节点的传输。传输次数的大小反映了数据传输量，次数越少说明转换策略的效率越高。假设 n_{di} 为数据 d_i 在布局前后的转换次数，显然 $N_t=\sum_1^N n_{di}$。为比较副本布局转换开销的大小，预先随机生成了任意两个节点之间单位数据的传输开销矩阵 $\boldsymbol{C}_{ij}=(c_{ij})_{n*n}$，其中 c_{ij} 是指数据中心 dc_i 和 dc_j 之间单位数据的传输开销；而数据中心中单位数据的删除开销相同。由此，布局转换开销是所有调整过程中的开销之和，包括传输以及删除等操作的开销。为了清晰地表述，传输次数和开销的大小均用相对值表示。

5.2　实验结果及分析

在仿真实验 1 中设置数据对象为 60 个，大小相同，均为 5 GB，所有数据中心的存储空间也相同，最多能放 6 个数据。X^{old} 与 X^{new} 均是随机产生的，但要求每个数据中心的数据对象在 X^{new} 与 X^{old} 布局下均不相同，即 $\forall dc_i((\bigcup_{X^{old}} d_i)\cap_{X^{new}} d_i)=\varnothing$。图 6 给出了在副本数目不同时各调度策略的数据传输次数与开销比较。



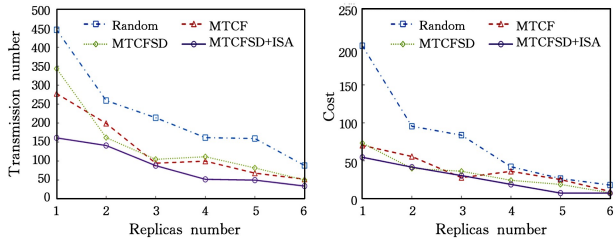
(a)　(b)

图 6　数据大小相同时各调度策略的数据传输次数与开销比较

Fig. 6　Transmission number and cost comparisons of different scheduling strategies with same data sizes

从图 6(a)中可以看出，随着数据副本的增加，在布局转换时数据传输的次数逐渐减少，其主要原因是可用的副本越多，可以越有效地减少跨数据中心的数据传输；同时，MTCFSD 策略相比其他策略而言，总的数据传输次数较少，而 Random 策略没有对任务调度进行合理优化，效率较低。同样地，从图 6(b)中可以看出，随着数据副本的增加，总的副本布局转换开销逐渐下降，但是 MTCFSD＋ISA 策略由于对队列进行了优化，传输次数少，其总的开销也是最低的。

仿真实验 2 与实验 1 的条件基本相同，主要区别在于实验 2 不再对数据对象的大小进行限定，而是服从 1～5 GB 的均匀分布。仿真实验结果如图 7 所示。



(a)　(b)

图 7　数据大小不同时各调度策略的数据传输次数与开销比较

Fig. 7　Transmission number and cost comparisons of different scheduling strategies with different data sizes

从图 7(a)中可以看出，尽管数据大小不同，但随着数据副本的增加，在布局转换时数据传输的次数仍然逐渐减少，而它们的布局转换开销也呈现出类似的趋势。总体来说，数据副本布局策略在保持较高的执行效率的同时，也具有较低的数据开销。

仿真实验 3 与实验 1 的条件基本相同，数据对象为 100 个，同时要求副本数为 5，比较不同数据中心存储空间大小对数据副本传输次数以及数据开销的影响，实验结果如图 8 所示。从图中可以看出，副本布局转换任务调度 MTCFSD＋ISA 与其他 3 种任务调度策略相比，在数据传输次数与数据开销等方面均具有较大优势。特别地，随着数据副本的增加，总的开销减小，当数据中心大小为 60 GB 时，MTCFSD＋ISA 策略的传输次数为 Random 策略传输次数的 1/3 左右，而开

销约为 1/2，说明任务调度队列优化进一步提升了调度的性能。

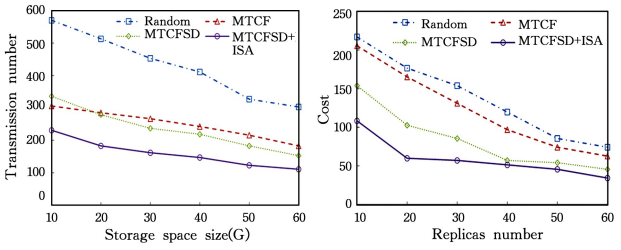


图 8 存储空间大小不同时各调度策略的数据传输次数与开销比较

Fig. 8 Transmission number and cost comparisons of different scheduling strategies with different storage space sizes

综上，不论在数据传输次数，还是在数据管理开销等方面，本文提出的数据副本布局转换策略都具有良好的性能，表明了该策略的有效性和可靠性。

结束语 云计算环境下的数据副本布局转换是一个关于数据副本迁移与删除任务的调度问题，而最小开销的数据副本布局转换则是 NP 完全问题。为此，从降低数据管理开销的角度，提出了 3 种副本布局转换策略，即最小传输开销优先(MTCF)、最大机会成本优先(MOCF)以及同数据最小传输成本优先(MTCFSD)。在转换可行性的基础上，通过对调度队列进行调整，提高了其有效性和实用性。本文的副本转换任务调度是一种串行任务调度，下一步的研究应从并行任务调度的角度研究数据副本布局转换任务调度算法。

参 考 文 献

[1] SUN X, LI Q Z, ZHAO P, et al. An optimized replica distribution method for peer-to-peer network[J]. Chinese Journal of Computers, 2014, 37(6): 1424-1434. (in Chinese)
孙新，李庆洲，赵璞，等. 对等网络中一种优化的副本分布方法[J]. 计算机学报，2014，37(6)：1424-1434.

[2] POLATO I, RÉ R, GOLDMAN A, et al. A comprehensive view of Hadoop research-a systematic literature review[J]. Journal of Network & Computer Applications, 2014, 46: 1-25.

[3] HAMROUNI T, SLIMANI S, CHARRADA F B. A survey of dynamic replication and replica selection strategies based on data mining techniques in data grids[J]. Engineering Applications of Artificial Intelligence, 2016, 48(C): 140-158.

[4] WU X. Data sets replicas placements strategy from cost-effective view in the cloud[J]. Scientific Programming, 2016(11): 1-13.

[5] SAADAT N, RAHMANI A M. PDDRA: A new pre-fetching based dynamic data replication algorithm in data grids[J]. Future Generation Computer Systems, 2012, 28(4): 666-681.

[6] YANG X L, QIAN C, ZHU F X. Evaluation method of big data service based on cloud computing[J]. Computer Science, 2018, 45(5): 295-299. (in Chinese)
阳小兰，钱程，朱福喜. 基于云计算的大数据服务资源评价方法[J]. 计算机科学，2018，45(5)：295-299.

[7] LIU W, PENG S, DU W, et al. Security-aware intermediate data placement strategy in scientific cloud workflows[J]. Knowledge & Information Systems, 2014, 41(2): 423-447.

[8] LI W, YANG Y, YUAN D. Ensuring Cloud data reliability with minimum replication by proactive replica checking[J]. IEEE Transactions on Computers, 2016, 65(5): 1494-1506.

[9] GRACE R K, MANIMEGALAI R. Dynamic replica placement and selection strategies in data grids-A comprehensive survey [J]. Journal of Parallel & Distributed Computing, 2014, 74(2): 2099-2108.

[10] WU X G. Minimum cost data replicas distribution based on dynamic programming in the cloud storage system[J]. Computer Engineering, 2017, 43(7): 29-37. (in Chinese)
吴修国. 云存储系统中基于动态规划的最小开销数据副本布局研究[J]. 计算机工程，2017，43(7)：29-37.

[11] LIU G, LI J, XU J. An Improved Min-Min Algorithm in Cloud Computing[C]// Proceedings of the 2012 International Conference of Modern Computer Science and Applications. Springer Berlin Heidelberg, 2013: 47-52.

[12] MA J, LI W, FU T, et al. A Novel Dynamic Task Scheduling Algorithm Based on Improved Genetic Algorithm in Cloud Computing[M]// Wireless Communications, Networking and Applications. Springer India, 2016: 184-186.

[13] TANG Y T, HUANG J, XIAO Q. Task scheduling algorithm for MapReduce based on DAG[J]. Computer Science, 2014, 41(S1): 42-46. (in Chinese)
唐一韬，黄晶，肖球. 一种基于 DAG 的 MapReduce 任务调度算法[J]. 计算机科学，2014，41(S1)：42-46.

[14] KHULLER S, KIM Y A, MALEKIAN A. Improved Approximation Algorithms for Data Migration[J]. Algorithmica, 2012, 63(1/2): 347-362.

[15] LU P, ZHANG L, LIU X, et al. Highly efficient data migration and backup for big data applications in elastic optical inter-data-center networks[J]. Network IEEE, 2015, 29(5): 36-42.

[16] ANDRONIKOU V, MAMOURAS K, TSERPES K, et al. Dynamic Qos-aware data replication in grid environments based on data "importance"[J]. Future Generation Computer Systems, 2012, 28(3): 544-553.

[17] ATENIESE G, BURNS R, CURTMOLA R, et al. Provable data possession at untrusted stores[C]// Proceedings of ACM Conference on Computer and Communications Security. New York: ACM Press, 2007: 598-609.

[18] LOUKOPOULOS T, TZIRITAS N, LAMPSAS P, et al. Implementing replica placements: feasibility and cost minimization [C]// Proceedings of Parallel and Distributed Processing Symposium. New York: IEEE Press, 2007: 1-10.

[19] CALHEIROS R N, RANJAN R, BELOGLAZOV A, et al. Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms[J]. Software Practice & Experience, 2011, 41(1): 23-50.