

一种云计算环境下大数据动态迁移策略

张晋芳,王清心,丁家满,刘彦君,黄 心

(昆明理工大学信息工程与自动化学院,昆明 650500)

摘 要: 云计算环境中大数据应用在数据迁移方面遇到各种问题,主要表现为如何在迁移过程中减少网络访问次数,减少全局时间消耗,以及提高效率的同时兼顾全局的负载均衡等。为此,对数据迁移进行建模,描述动态迁移策略,分别针对策略中的全局时间消耗、网络访问次数和全局负载均衡 3 个参数进行求解,并在云计算仿真平台 Cloudsim 下进行实验。结果表明,使用数据动态迁移策略后,任务完成时间比 Zipf 分布减少约 10%,网络访问次数低于原始 Zipf 分布并趋于稳定;全局负载均衡方面,节点存储空间方差趋于 0。

关键词: 云计算;大数据;负载均衡;数据迁移;网络访问;数据集

中文引用格式:张晋芳,王清心,丁家满,等. 一种云计算环境下大数据动态迁移策略[J]. 计算机工程,2016,42(5):13-17.

英文引用格式:Zhang Jinfang, Wang Qingxin, Ding Jiaman, et al. A Big Data Dynamic Migration Strategy in Cloud Computing Environment[J]. Computer Engineering, 2016, 42(5):13-17.

A Big Data Dynamic Migration Strategy in Cloud Computing Environment

ZHANG Jinfang, WANG Qingxin, DING Jiaman, LIU Yanjun, HUANG Xin

(Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China)

[Abstract] Big data applications meet various challenges in data migration in cloud computing environment. It mainly manifests in below aspects: reduce the number of network access, reduce the overall time consumption and improve the efficiency by the time of balancing the global load in the migration process and so on. Facing these challenges, it builds the problem model and describes the dynamic migration strategy, then solves the global time consumption of data migration, the number of network access and global load balance in these three parameters. The cloud computing simulation experiment is done under Cloudsim experimental platform. The result shows that the proposed data dynamic migration strategy makes the task completion time reduced by 10% than Zipf distribution, network access number be lower than Zipf and tends to be stable. And in global load, the variance of the node's store space is closed to zero.

[Key words] cloud computing; big data; load balance; data migration; network access; dataset

DOI:10.3969/j.issn.1000-3428.2016.05.003

1 概述

随着人类产生、获取和存储数据能力的飞速发展,大数据应用已广泛出现在天文、物理、生物信息和经济等领域。作为一种新型网络计算模式,云环境正逐步成为支撑大数据应用的基础架构之一^[1]。对于大数据,尚未有一个公认的定义,在目前的定义中,比较有代表性的是 3V 定义,即认为大数据需满足 3 个特点:规模性,多样性和高速性^[2]。大数据应

用在利用云计算环境的过程中遇到了一些新的挑战,特别是在数据迁移方面尤为突出。在云计算环境下部署并执行大数据应用,需要多数据中心的协作。并且用户对大数据的需求是变化的,使得某些数据中心空闲,而某些数据中心的数据却被频繁访问,甚至引起堵塞现象^[3]。这就需要对这些数据进行迁移,以提高数据中心的访问效率。

在多数数据中心环境下如何对大数据进行迁移变得尤为重要,具体表现为:

基金项目:国家自然科学基金资助项目(51467007);云南省应用基础研究计划基金资助项目(2013FZ020)。

作者简介:张晋芳(1989-),女,硕士研究生,主研方向为云计算、大数据;王清心(通讯作者),教授;丁家满,副教授;刘彦君、黄 心,硕士研究生。

收稿日期:2015-06-09

修回日期:2015-07-21

E-mail:yunnan403@sina.com

(1) 在云计算中,对大数据的需求是动态变化的。因此,需要对数据集进行迁移。在迁移的过程中,不可避免地要进行跨数据中心的数据传输,此过程中造成的时间开销成为一个亟待解决的问题。

(2) 在数据动态迁移的过程中,增加了网络的访问次数,如何在迁移过程中,尽量减少网络访问次数,成为需要考虑的一个问题。

(3) 在数据动态迁移过程中,要充分利用系统资源,并且保证系统负载均衡。数据中心节点间的负载问题也成为了一个需要考虑的因素。

针对这 3 个问题,本文提出云计算环境下大数据动态迁移策略,在完成任务时,跨数据中心的时间开销和网络访问次数均较少且整个系统负载均衡。

2 相关工作

云计算下的大数据动态迁移成为日益关注的问题^[4]。文献[5]针对云计算环境下数据密集型应用,提出一种面向流程的数据密集型应用的数据迁移方案。文献[6]针对单一任务类型提出一种基于时间期限和预算的调度算法。文献[7]针对独立任务包提出一种面向 Bot 应用的启发式数据选择策略,以期在完成任务花费的时间和经济成本之间取得平衡。但文中讨论的是独立任务包,即认为各个任务是相互独立的,不适用于带有相互依赖关系的数据密集型应用。文献[8]针对结构化数据利用数据网格对分布存储资源中的海量数据集进行访问、移动和修改。上述策略均是针对单一任务类型,不能满足云计算环境中作业类型负载多样且约束多元的需求。

文献[9]运用云计算模型与马尔可夫过程理论,建立云计算下的负载转移模型,给出转移概率算法来解决节点的负载失衡问题。文献[10]针对负载均衡提出了一种双限定值的虚拟机动态迁移的调度策略。文献[11]引用虚拟节点,提出一种基于节点容量感知的数据分配策略。这 3 种分布策略在负载均衡的性能上具有优势,但未考虑数据迁移过程中有可能带来的网络访问次数和时间消耗。

文献[12]给出基于相关度的两阶段高效数据放置策略和任务调度策略。文献[13]提出了一种基于聚类矩阵的数据布局策略,该方法利用 BEA^[14]算法得到聚类矩阵,然后基于聚类矩阵对所有数据集组成的集合进行划分。以上两种策略能有效减少跨数据中心科学 workflow 执行时的数据传输量和数据传输的总次数,但忽略了不同数据量的大小,不同的网络带宽等造成的时间消耗都不同。

综上所述,虽然针对云数据管理^[15]的研究比较多,但是关于云计算环境下大数据型应用的数据迁移问题研究得较少,当前少数针对该问题的研究也存在一定的局限性。

3 问题描述与建模

定义 1 云计算环境表示为 N 个分布在不同位置的数据中心组成的集合 $DC = \{dc_1, dc_2, \dots, dc_n\}$ 。这些数据中心之间通过不同的网络连接,而这些网络中的带宽是不相同的。其中, dc_n 为编号为 n 的数据中心。真实的物理网络可以将它抽象为逻辑上的网络拓扑结构。 DC 中各个数据中心间的网络带宽用矩阵表示如下:

$$B = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1|DC|} \\ b_{21} & b_{22} & \dots & b_{2|DC|} \\ \vdots & \vdots & & \vdots \\ b_{|DC|1} & b_{|DC|2} & \dots & b_{|DC||DC|} \end{bmatrix}$$

对于所有的 i, j , 当且仅当 $i \neq j$ 时, b_{ij} 表示的是数据中心 dc_i 和 dc_j 之间的网络带宽值。

定义 2 存储在数据中心的一个数据集可以表示为二元组 $d_i = \langle ds_i, lc_i \rangle$ 。其中, ds_i 表示数据集 d_i 的大小; lc_i 表示 d_i 的存储位置, 且 $lc_i \in DC^{[10]}$ 。

定义 3 大数据型应用可以表示为二元组 $P = (J, D)$ 。其中, J 表示构成任务 P 的 n 个任务的集合; D 是执行这些任务所需数据集的集合。

定义 4 设源数据中心、目标数据中心和将 d 从源数据中心传输到目标数据中心的时间开销分别为: dc_i, dc_j 和 $T(d, dc_i, dc_j)$, 则 $T(d, dc_i, dc_j)$ 可以表示为:

$$T(d, dc_i, dc_j) = ds/BW(link(dc_i, dc_j)) + C_{ij} \quad (1)$$

其中, $ds/BW(link(dc_i, dc_j)) + C_{ij}$ 表示的是将 d 从 dc_i 传输到 dc_j 所需要的实际时间开销。另外, 在进行跨数据中心的数据传输过程时, 还存在着请求、响应、建立连接、断开连接等步骤, 也会造成一部分时间开销, 将这部分时间开销记为 C_{ij} 。因为在云计算环境下面向大数据应用的数据规模十分巨大, 相比较而言, C_{ij} 很小, 所以将 C_{ij} 代表的与数据集大小无关的时间开销忽略。因此, 式(2)可用来近似计算单个数据集单次跨数据中心传输所造成的时间开销:

$$T(d, dc_i, dc_j) \approx ds/BW(link(dc_i, dc_j)) \quad (2)$$

4 数据动态迁移策略

数据迁移方案如图 1 所示, 可以分为以下 3 步:

(1) 根据数据中心的负载,求得负载较少的数据中心作为目标数据中心集。

(2) 判断数据集从源数据中心到各个目标数据中心的时间消耗,并将目标数据中心按时间消耗递

增排序。

(3) 根据当前网络中的网络访问次数与需要转移的数据量,定义目标数据中心的数量范围,并根据 Threshold 得出最终的目标数据中心。

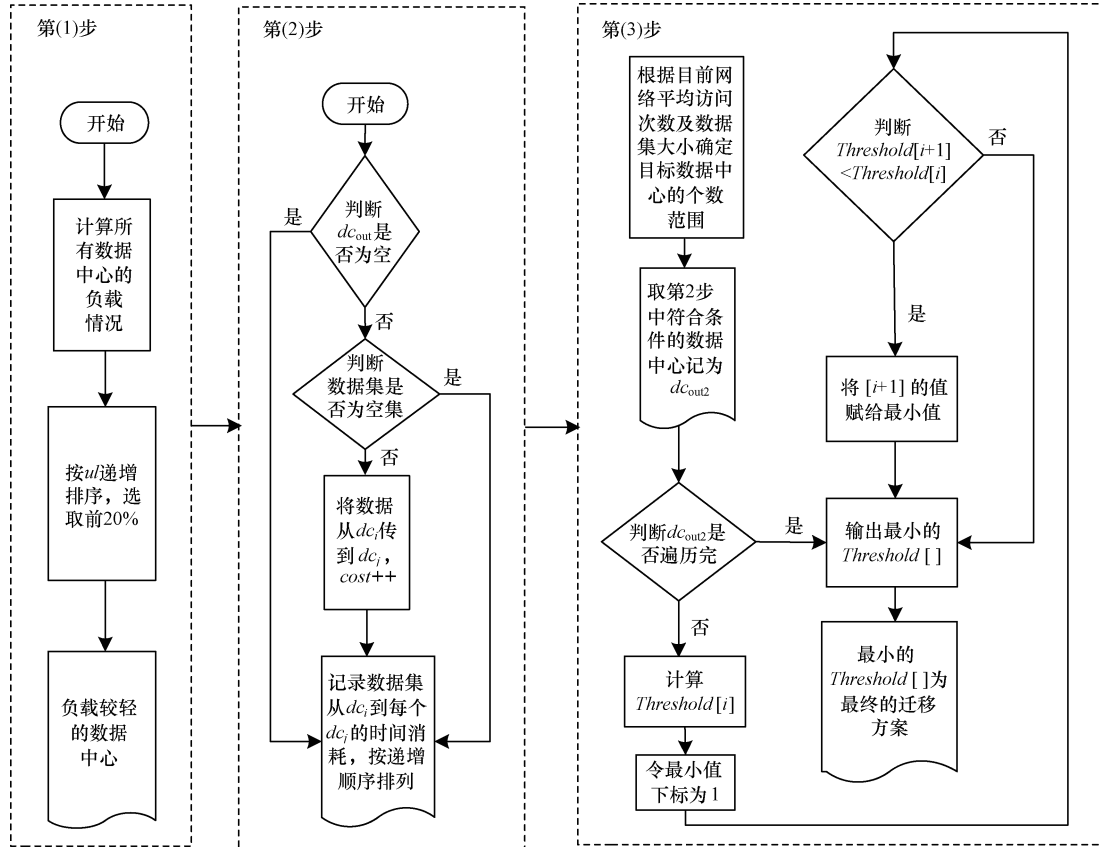


图1 迁移方案流程

4.1 负载较轻的数据中心

只有当某一数据中心负载过重时,才需要数据迁移。记录每个数据中心的负载信息,包括负载能力、实际负载等。 $f_z(dc_i, t)$ 表示数据中心 dc_i 的负载能力,则负载 $f_z(dc_i, t) = (f_z, f_{z_h}, f_{z_l})$ 。其中, f_z, f_{z_h}, f_{z_l} 分别表示当前负载、负载上限、负载下限。

目标数据中心集:将全网中工作负载最轻的20% 数据中心作为目标数据中心,记为 DC_{out} 。即: $DC_{out} = \{x | x \text{ 不属于存在 } d_i \text{ 的数据中心, 且当前数据中心的工作负载满足 } f_z(dc_i, t) < f_{z_h}\}$ 。

数据中心的理想负载情况是 $f_{z_l} < f_z < f_{z_h}$, 如果 $f_z \geq f_{z_h}$, 服务节点处于过载状态; 否则, 认为该服务节点过于空闲。分别定义数据中心的过载指数 ol 和欠载指数 ul :

$$ol(dc_i, d_i, t) = (c_{out}(dc_i, d_i, t) - f_{z_h}(dc_i)) / f_{z_h}(dc_i) DC_{out} \quad (3)$$

$$ul(dc_i, d_i, t) = (f_{z_l}(dc_i) - c_{out}(dc_i, d_i, t)) / f_{z_l}(dc_i) \quad (4)$$

负载情况的四分位点求解算法见算法1。

算法1

输入 当前网络中各个数据中心 dc_i 的负载 f_z , 网络中数据中心个数 N

输出 网络中负载最小的前20% 的数据中心

for $i = 1$ to N ;

for $j = n$ to $i + 1$;

if $f_{z[j]} < f_{z[i]}$;

exchange $f_{z[j]}$ and $f_{z[i+1]}$;

endif;

选择 front 20% 数据中心作为目标数据中心;

4.2 时间消耗的求解

数据迁移方案求解的第二阶段, 计算需要迁移的数据集从源数据中心到达每个目标数据中心的时间消耗, 并将目标数据中心按时间消耗递增排序, 具体算法见算法2。

算法2

输入 dc_i, dc_{out}, d

输出 此时的时间近似开销 $TimeCost(d, dc_i, dc_j)$

```

cost=0,init data[];
i=data[0],delete i from data[];
While(dcout!=null)
    while(data[]!=null) Timecost++;
    compute TimeCost(d,dci,dcj)[];
    //为数据集到每个数据中心的时间消耗
    min TimeCost(d,dci,dcj)=TimeCost(d,dci,dcj)[0];
    if TimeCost(d,dci,dcj)[i+1]<TimeCost(d,dci,dcj)[i]
        TimeCost(d,dci,dcj)=TimeCost(d,dci,dcj)[i];
    printf TimeCost(d,dci,dcj);

```

4.3 Threshold 较小的目标数据中心

记录迁移过程中,带来的网络访问次数为 $InternetVisit$,则:

$$InternetVisit \leq \sum_i^n IV_i/T \quad (5)$$

在式子 $\sum_i^n IV_i/T$ 中, T 表示到目前为止总的时
间; $\sum_i^n IV_i$ 表示将总时间均分为 n 等分,记录每一等
分时间内,云计算环境中的网络访问次数为 IV_i 并求
和,则可以确定目标节点的数量 N 为:

$$N \leq \sum_i^n IV_i/T \quad (6)$$

取 DC_{out} 中时间消耗较少的前 N 个数据中心
记为 dc_{out2} (按照时间消耗递增排序)。目标数据中心
个数的增加或者减少,会引起时间消耗和负载的变
化。因此,需要在这 3 个目标之间取得一个平衡。

$$Threshold = \alpha Timecost + \beta fz + \gamma InternetVisit \quad (7)$$

算法 3

输入 dc_{out2}

输出 $Threshold\{\}$

```

if(i=1,i<dcout2,i++)
    //i 代表从 dcout2 中选取前几个
    compute Threshold{i};
    minThreshold{ } = Threshold{i};
    if Threshold{i+1} < Threshold{i}
        minThreshold{ } = Threshold{i};
    endif
    printf Threshold{ }
//最后的 Threshold 里保存的数据中心数即为最后的
//迁移策略

```

5 仿真实验与分析

5.1 实验设置与环境

实验是在 CloudSim 的模拟平台上完成的。本
文创建 6 个数据中心(数据中心的配置如下:虚拟机

CPU 为四核,主机内存 16 GB,带宽 1 000 MIPS,镜
像大小 1 000 000 M,处理能力 10 MIPS),这些数据
中心间通过高速且不同带宽的网络连接。执行一个
大数据型应用需要 5 个数据集,执行任务之前,这些
数据集已经根据 Zipf 分布^[13] 而分布存储在这 6 个
数据中心中。实验中数据集初始分布为 Zipf,数据
集第二次分布为本文的数据迁移方案,每个任务所
需的数据集为 5,任务量为[100,1 000],以 100 为单
位长度。

实验以任务量为变化参数。比较使用迁移方案
后和原始数据分布的性能。性能包括任务完成时
间、数据传输时间占总执行时间的比例、负载均衡程
度以及网络访问次数。所有实验数据都是 10 次实
验的平均值。

5.2 实验结果与分析

如图 2 所示,与 Zipf 分布相比,本文的数据迁移
方案使得任务完成时间减少约 10%。

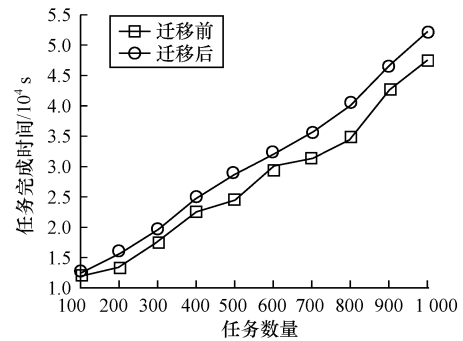


图 2 任务完成时间

如图 3 所示,经过本文的数据迁移方案,完成任
务时,数据传输时间占总执行时间的比例减少。当
任务量达到 400 后,相比于 Zipf 优势更为明显。

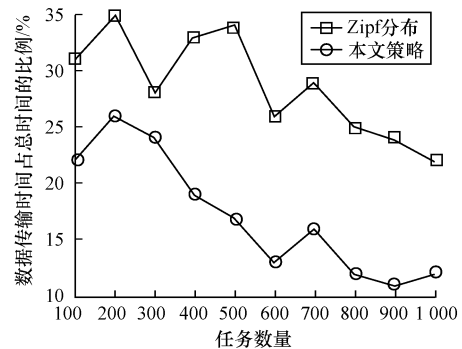


图 3 数据传输时间占总时间的比例

如图 4 所示,动态迁移策略中节点资源分配的
均匀程度均优于原始的 Zipf 分布,并且随着物理节
点数的增加,动态分布策略的节点存储空间方差值
收敛于 0。

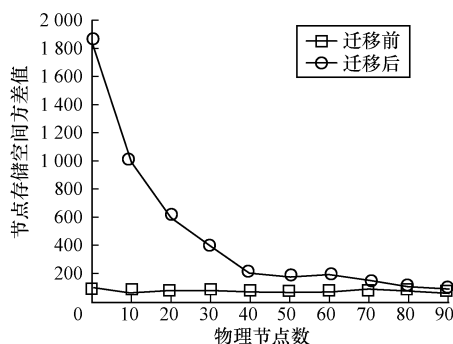


图4 原始算法与迁移算法负载均衡性能比较

如图5所示,动态迁移策略中网络访问次数均少于原始的 Zipf 分布,并且随着任务数量的增加,动态分布策略网络访问次数变化趋于稳定。

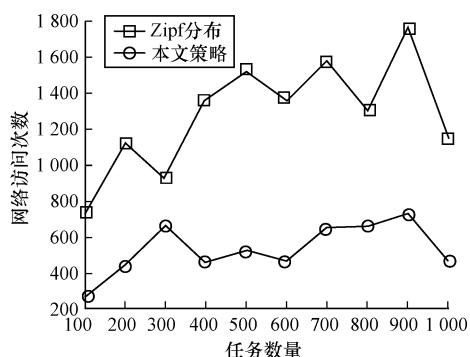


图5 原始算法与迁移算法网络访问次数比较

6 结束语

针对云计算环境下大数据的数据布局问题,本文通过对环境及应用的分析,提出数据动态分布策略,得到数据动态迁移算法,并在云模拟器 CloudSim 上进行仿真实验。通过与原始数据分布的对比表明,该策略使得任务执行效率明显提高。

目前在数据迁移时,假定节点间访问成本是相同的,但云计算按需付费的特点存在从不同节点访问相同数据集需要支付不同费用的情况,因此,未来计划运用其他的方法解决这一问题。

参考文献

[1] Labrinidis A, Jagadish H V. Challenges and Opportunities with Big Data[J]. Proceedings of the VLDB Endowment, 2012, 5(12): 2032-2033.

- [2] 秦秀磊, 张文博, 魏峻, 等. 云计算环境下分布式缓存技术的现状与挑战[J]. 软件学报, 2013, 24(1): 50-66.
- [3] Abirami S P, Shalini R. Linear Scheduling Strategy for Resource Allocation in Cloud Environment[J]. Journal on Cloud Computing: Services and Architecture, 2012, 2(1): 9-17.
- [4] Dean J, Ghemawat S. MapReduce: Simplified Data Processing on Large Clusters[J]. Communications of the ACM, 2008, 51(1): 107-113.
- [5] 郑湃, 崔立真, 王海洋, 等. 云计算环境下面向数据密集型应用的数据布局策略与方法[J]. 计算机学报, 2010, 33(8): 1472-1480.
- [6] 刘亚秋, 邢乐乐, 景维鹏. 云计算环境下基于时间期限和预算的调度算法[J]. 计算机工程, 2013, 39(6): 56-59.
- [7] 杜薇, 崔国华, 刘伟, 等. 云环境下面向数据密集型应用的数据选择策略研究[J]. 计算机科学, 2012, 39(6): 30-34.
- [8] Venugopal S, Buyya R. An SCP-based Heuristic Approach for Scheduling Distributed Data-intensive Application on Global Grids[J]. Journal of Parallel and Distributed Computing, 2008, 68(4): 471-487.
- [9] 刘之家. 一种基于云计算的负载均衡技术的研究[J]. 广西师范学院学报: 自然科学版, 2011, 28(2): 93-96.
- [10] 方义秋, 葛道红, 葛君伟. 云计算环境下基于虚拟机动态迁移的调度策略研究[J]. 微电子学与计算机, 2012, 29(4): 45-48.
- [11] 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.
- [12] 刘少伟, 孔令梅, 任开军, 等. 云环境下优化科学 workflows 执行性能的两阶段数据放置与任务调度策略[J]. 计算机学报, 2011, 34(11): 2121-2130.
- [13] Yuan Dong, Yang Yun, Liu Xiao, et al. A Data Placement Strategy in Scientific Cloud Workflows[J]. Future Generation Computer Systems, 2010, 26(8): 1200-1214.
- [14] McCormick W T, Schweitzer P J, White T W. Problem Decomposition and Data Reorganization by a Clustering Technique[J]. Operations Research, 1972, 20(5): 993-1009.
- [15] 刘正伟, 文中领, 张海涛. 云计算和云数据管理技术[J]. 计算机研究与发展, 2012, 49(z1): 26-31.

编辑 顾逸斐