

# 网络 GIS 中最佳负载均衡的分布式缓存副本策略

李 锐<sup>1,2</sup> 唐 旭<sup>3</sup> 石小龙<sup>1</sup> 樊珈珮<sup>1</sup> 桂志鹏<sup>2,4</sup>

1 武汉大学测绘遥感信息工程国家重点实验室,湖北 武汉,430079  
2 地球空间信息技术协同创新中心,湖北 武汉,430079  
3 武汉大学资源与环境科学学院,湖北 武汉,430079  
4 武汉大学遥感信息工程学院,湖北 武汉,430079

**摘 要:**云环境下的网络地理信息服务具有分布性和异构性,空间数据(瓦片)的访问请求具有高度聚集性和不均匀性。以最小化负载不均衡度为目标,提出了一种应用于异构的、分布式高速缓存集群系统的多副本策略。该策略针对瓦片访问请求存在不均衡性,最小化热点访问数据的通信权重值,最大化地利用分布式集群缓存能力生成副本;针对异构集群环境下服务器处理能力的不均衡性,根据服务器性能和瓦片副本的通信权重值,匹配各个服务器的缓存能力部署副本。实验证明,该策略避免服务器拥塞的同时,能充分利用有限的分布式集群缓存能力,实现较好的负载均衡和较高的资源利用率,并能获得良好的缓存命中和请求响应性能。  
**关键词:**负载均衡;集群;异构;缓存;服务质量  
**中图法分类号:**P208;TP311      **文献标志码:**A

地理信息服务从专业化到大众化的转变,应用环境由桌面形式向网络形式的转变,庞大的数据量和海量的访问量使其网络流量急剧增加,导致网络拥塞和服务器过载,最终影响用户的服务质量。云环境下的网络地理信息服务质量问题,即如何支持大规模、高强度的并发用户访问,以提供高可靠、高可用、可扩展的地理信息服务<sup>[1]</sup>,成为学术界和工业界的关注热点。而空间云计算服务资源的分布性、异构性、动态性,使得其网络服务环境比一般的地理信息服务环境更为复杂<sup>[2]</sup>。分布式缓存多副本策略多用于云计算中分布式数据网格(即分布式高速缓存),为大容量、大规模的地理信息共享实现并行高效的服务。其中副本的生成与部署直接影响空间云计算中的网络地理信息服务性能与负载均衡效率。

目前,越来越多的研究关注用户访问与服务系统的交互作用,多基于数据的访问统计概率建立副本生成与部署机制。文献[3]提出了面向访问热度的同构服务节点副本生成/放置算法——MMPacking。Dynamic Repacking 算法<sup>[4]</sup>以 MMPacking 算法为基础,从理论上解决了动态预测访问概率生成和更新副本问题。Ibarkai<sup>[5]</sup>提

出,若基于数据的访问概率生成副本对象个数,可达到最佳的副本效果,并提出了经典的资源分配算法 Adams' Monotone Divisor Replication (AMDR)。Optimal Replica Strategy(ORS)<sup>[6]</sup>根据集群服务器总数将缓存数据划分为  $N$  个区域,并根据数据的流行度分配数据副本至对应的区域。文献[7]依据数据访问服从 Zipf 分布,将流行度高的数据以迭代放置方法和轮盘赌方式随机选择服务器,进行副本生成和缓存放置。在单纯的副本放置算法上,Dan<sup>[8]</sup>提出了基于服务器网络带宽的副本放置算法;文献[9]提出了满足 QoS(quality of service)需求的副本放置算法。以上研究多基于数据对象的访问热度划分数据对象,进行副本的生成与放置,且基于同构的分布式服务器。

文献[10,11]表明,网络地理信息服务中空间数据对象(瓦片)的访问服从重尾分布,20%的瓦片吸引了 80%的请求,访问具有局部性和倾斜性。它反映出一种大规模用户访问瓦片数据的群体行为,具有高度聚集性,也体现了瓦片数据的访问热度,具有不均衡性。并且,瓦片金字塔模型的空间数据管理方法为数据的缓存提供了良好的支

持,可以将瓦片数据作为访问粒度和缓存粒度。云环境下的网络地理信息集群服务是分布的、异构的,集群服务器的处理能力也是不均衡的。鉴于此,本文提出了一种最佳负载均衡的分布式缓存多副本生成与放置策略。

# 1 面向负载均衡的分布式缓存多副本网络模型

## 1.1 网络模型

分布式高速缓存服务具有高性能、可伸缩性和一致性等特征,多采用独立于应用服务且可被独立扩展的架构。本文建立了基于排队论的异构分布式集群缓存网络模型,如图 1 所示,用于描述面向负载均衡的分布式缓存多副本生成与放置策略。图 1 中, $S=(S_1, S_2, \dots, S_N)$ 表示异构分布式集群缓存系统中的  $N$  个服务器。由于各个服务器在存储空间、服务器的处理能力等方面各异,用  $V_i$  表示服务器  $S_i$  的处理性能值,即每秒能处理的服务请求数; $C_i$  表示服务器  $S_i$  的缓存空间大小,即缓存瓦片个数的能力; $t=(t_1, t_2, \dots, t_M)$ 表示  $M$  个不同的缓存瓦片数据对象; $p_i$  表示瓦片对象  $t_i$  的热度值,即访问概率; $b_i$  表示其存储大小。

网络地理信息服务中瓦片访问请求到达率可描述为泊松事件<sup>[12]</sup>,而瓦片访问请求在服务器端的处理时间具有无记忆性,可用负指数分布来描述。因此,图 1 网络模型中用户访问请求到达率服从均值为  $\lambda$  的泊松分布,请求到达间隔服从均值为  $1/\lambda$  的负指数分布,服务器处理请求时间服从负指数分布,均值为  $1/\mu$ 。图 1 中负载均衡器根据副本策略和瓦片请求任务转发策略,按照概率  $p_i$  将请求分发给服务器  $S_i$  处理。当请求到达服务器  $S_i$  时,若服务器空闲,则被处理,否则进入等待队列。根据排队论和瓦片访问请求特征,模型中的队列为一个 M/M/S/ $\infty$  排队模型。由于服务请求处理时间中考虑了排队等待时间,采用无限长队列。用户请求的响应时间包括请求处理时间和响应数据在链路上的传输时延。请求的处理时间为服务器队列中排队等待的时间和被服务器处理的服务时间之和。根据文献[12],其值是服务器处理能力的函数。在网络链路时延变化不大的情况下,如果能最小化请求处理时间,则能获得较短的请求响应时间。在本文提出的副本策略中,具有高访问负载的热点瓦片的副本个数及其在不同处理能力的服务器上的分布是影响请求响

应时间的关键因素。

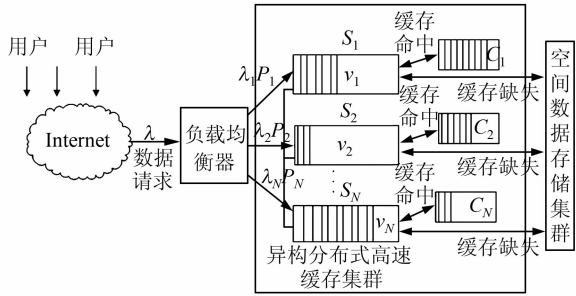


图 1 异构分布式集群高速缓存网络模型

Fig. 1 Network Model of Heterogeneous Distributed Cluster-based High-speed Caching for Load-balancing

## 1.2 负载均衡优化目标

副本生成与放置策略目标是提高地理信息网络服务的质量和服务的可得性。一般而言,副本生成是加强瓦片对象的可得性,通过增强瓦片对象的副本程度(瓦片对象的副本平均个数)实现。而副本放置直接影响分布式集群缓存系统的负载均衡,若服务器负载不均衡,将导致瓦片对象的访问可得性减小。因此,分布式集群缓存系统的负载均衡是副本策略优化的主要目标之一。用  $L$  表示分布式缓存集群系统中服务器之间的不均衡度, $r_i$  表示瓦片对象  $t_i$  的副本个数,则副本策略的优化目标公式为:

$$Obj = \alpha \cdot \sum_{i=1}^M r_i / M - \beta \cdot L \tag{1}$$

式中, $\alpha$  和  $\beta$  是副本程度与服务器负载之间关系的权重值; $L$  表示负载不均衡度,它有多种计算方式,经典的计算公式为:

$$L = \max_{V, S_i \in S} | l_i - \bar{l} | \tag{2}$$

式中, $l_i$  表示服务器  $S_i$  的当前负载程度; $\bar{l}$  表示分布式集群缓存系统的平均负载值,即

$$\bar{l} = \sum_{i=1}^N l_i / N$$

从式(1)可见,副本策略的总体目标可分解为增强副本程度和服务器负载均衡两个子目标,分别由副本生成策略和副本放置策略实现,但它们之间又存在着相辅相成的关系。副本生成策略指尽可能多地生成副本;副本放置策略指在负载均衡的前提下,尽可能地将生成的副本全部放置,最终达到式(1)的目标。

为实现增强副本程度目标,假设副本生成过程中使用分布式集群缓存系统中的全部,即最大化地使用缓存空间,同时,假设副本生成过程中也需考虑到副本放置。本文提出通过式(3)和式(4)

实现最大化使用缓存空间的同时简洁副本放置过程,以达到分布式集群缓存系统的负载均衡。

$$\text{Minimize } \max_{\forall w_i \in w} \{w_i\} \quad (3)$$

$$w_i = w_j, \forall t_i, t_j \in t, 1 \leq i, j \leq M, i \neq j \quad (4)$$

式中,  $w_i$  表示瓦片  $t_i$  副本的网络通信权重值,在本文中用其访问概率表示,即  $w_i = p_i/r_i$ 。

式(3)表示尽量使网络访问概率高的瓦片其副本的网络通信权重值最小化,即使访问量大的瓦片尽量分配多副本,并使其每个副本的访问负载最小化。为实现负载均衡目标和简洁副本放置过程,提出式(4)进一步实现式(3)。

式(4)表示每个瓦片的副本网络通信权重值相等。在同一性能的服务器组成的分布式集群缓存系统中,可以通过随机放置副本即可达到最佳的负载均衡。但对于图 1 中的异构的分布式缓存多副本网络模型,副本的生成和放置策略受限于以下三个条件:① 每台服务器的缓存空间大小为  $C_i$ ;② 每台服务器的处理能力为  $V_i$ ;③ 同一副本不能放置在同一服务器中。

若  $p_i$  表示相应瓦片对象  $t_i$  的热度值,  $b_i$  表示其存储大小,使  $\pi(t_i) = k$  表示瓦片  $t_i$  的一个副本被放置于服务器  $S_k$  中,则限制条件①可表示为:

$$\sum_{\pi(t_i)=k, \forall t_i \in t} b_i \leq C_k \quad (5)$$

用  $\lambda$  表示访问高峰期间请求的到达率,则限制条件②可表示为:

$$\sum_{\pi(t_i)=k, \forall t_i \in t} w_i \cdot \lambda \leq V_k \quad (6)$$

根据限制条件③,定义每个瓦片的副本对象个数不超过服务器个数  $N$ ,即

$$1 \leq r_i \leq N \quad (7)$$

### 1.3 理想的负载均衡副本策略

依据目标函数式(1)、式(2)和限制条件式(5)、式(6)、式(7),可得出如下副本生成的方法:设分布式集群系统的缓存能力之和为  $C = \sum_{k=1}^N C_k$ ,缓存瓦片对象的热度值之和为  $P = \sum_{i=1}^M p_i$ ,则每个缓存瓦片的网络权重均值为  $\bar{w} = P/C$ 。为实现式(4)的负载均衡目标,瓦片对象  $t_i$  的副本个数  $r_i$  可通过式(8)得出:

$$r_i = p_i/\bar{w} \quad \forall t_i \in t, 1 \leq i \leq M \quad (8)$$

且基于条件③的限制  $1 \leq r_i \leq N$ 。

因该方法依据集群系统总缓存容量生成副本,则其在理论上也实现最大化的副本个数。而副本需要在异构服务器组成的分布式集群缓存系统中进行合理放置,以达到最佳的负载均衡。一

般来说,负载均衡目标是根据各服务器的性能分配工作负载,最大限度地利用集群的优势并提供更好的网络服务质量。在图 1 的网络模型中,  $V_i$  值大,则服务器性能好。根据排队论,  $\rho_i = \lambda_i/V_i$  可用于表示服务器  $S_i$  的服务强度,其值反映系统的繁忙程度。用  $W_i$  表示服务器  $S_i$  上缓存副本的总被访问率,则服务器  $S_i$  的请求到达率为  $\lambda_i = \lambda \cdot P_i = \lambda \cdot W_i$ 。从图 1 可见,服务器  $S_i$  达到平衡时的负载强度为  $\rho_i = \lambda \cdot W_i/V_i$ 。因此,如果分布式集群缓存系统处于负载均衡状态,则每个集群的服务器达到式(9)的目标即可。而对于服务器  $S_i$ ,其服务性能  $V_i$  是固有的。因此,可以根据式(9)获取每个服务器处于平稳状态时的被访问概率  $W_i$ ,继而再将式(8)生成的副本进行放置,则可以使服务器在最好的性能下工作,且分布式集群缓存系统达到负载均衡。

$$\frac{W_1}{V_1} = \frac{W_2}{V_2} = \dots = \frac{W_N}{V_N} = \bar{l} \quad (9)$$

综上所述,若依据式(3)、式(4)、式(8)实现副本生成,依据式(9)进行副本放置,则可实现根据服务器的性能与缓存空间大小的关系最大化生成副本,并进行最优副本放置。产生的副本策略则可达总体负载均衡优化目标式(1),并可实现最小化式(2)中的负载不均衡度。

## 2 异构分布式集群缓存系统中的多副本策略

式(4)和式(9)是理想的副本生成和放置方法。对于同构分布式集群缓存系统来说,很容易实现负载均衡目标。而对于异构系统,必须考虑集群的服务器性能的异构性和缓存容量的异构性。因此,基于式(7)、式(8)、式(9),提出最佳负载均衡的副本策略(optimal average load replication strategy, OAL),最大化副本个数(数据访问服务可得性)的同时,实现异构系统中副本放置的负载均衡。

### 2.1 缓存多副本生成方法

因热点瓦片对象被访问频繁,其热度值较大。根据式(8)计算时,它的副本个数往往大于  $N$ ,则其副本的通信权重值将大于  $\bar{w}$ 。根据多副本限制条件式(7),瓦片对象的副本个数最大值为  $N$ ,将导致缓存得不到有效的利用。鉴于此,本文提出基于式(7),通过逐步减少副本的平均网络通信权重值,实现瓦片副本间接接近均衡的网络权重,同时生成更多的瓦片副本进行缓存,并简洁负载均衡

的副本放置过程。

缓存多副本生成方法如下。首先基于式(7)和式(8)计算各个瓦片的副本个数。对于副本个数大于服务器个数的瓦片,将它的副本个数设置为  $N$ 。然后,为了达到最大化副本个数目标,基于式(8)和剩余的缓存空间,再次计算目前副本个数不为  $N$  的瓦片对象的副本个数。循环此过程,直至不再有个数大于  $N$  的瓦片副本产生,并且剩余的瓦片产生的副本网络通信权重值相等。如此一来,最后获取副本的瓦片有两种类型:瓦片拥有  $N$  个副本;未拥有  $N$  个副本的瓦片都拥有相等网络通信权重值的副本。

具体步骤如下:

1) 设  $M_{\text{left}}=M, C_{\text{left}}=C$ , 其中,  $M_{\text{left}}$  表示当前未生成副本的瓦片对象个数,  $C_{\text{left}}$  为集群系统剩余的缓存空间;

2) 将全部瓦片对象的热度值相加,  $P = \sum_{i=1}^M p_i$  且  $P_{\text{left}}=P, P_{\text{left}}$  为当前未生成副本的缓存瓦片对象的热度值总和, 集群系统总共能缓存的副本个数  $C = \sum_{k=1}^N C_k$ , 计算网络权重均值  $\bar{w} = P/C$ ;

3) 根据  $\bar{w}$  和式(8)计算各个瓦片的副本个数  $r_i$ ;

4) 若瓦片  $t_i$  的副本  $r_i > N$ , 则置  $r_i = N$ , 记录副本为  $N$  的瓦片个数为  $a$ ;

5) 重新计算网络通信权重均值。剩余缓存可分配的副本个数为  $C_{\text{left}} = C_{\text{left}} - a \cdot N$ , 热度值总和为  $P_{\text{left}} = P_{\text{left}} - \sum_{i=1}^a p_i$ , 计算权重均值  $\bar{w} = P_{\text{left}}/C_{\text{left}}$ ;

6) 对剩余的  $M_{\text{left}} = M_{\text{left}} - a$  个瓦片对象继续采用式(8)计算对应的副本个数, 若  $r_i > N$ , 循环第 4)~6) 步, 直至未出现  $r_i > N$ 。剩余的瓦片对象的副本个数为最后一次根据式(8)产生的副本数, 其权重值为  $\bar{w}$ 。

该副本生成方法根据服务器个数以及分布式集群缓存大小, 通过逐步减小平均通信权重值, 实现瓦片副本通信权重均衡最优的目标。与此同时, 不但保证了访问热度高的瓦片具有最大化或更多的副本, 而且通过降低平均通信权重值, 热度低的瓦片也可产生副本, 从而最大化地使用缓存空间, 提高系统资源的利用率。如此达到了式(1)中的副本可得性目标, 即负载均衡的前提下, 尽可能最大化地产生副本个数, 且算法的复杂度为

$O(M)$ 。

2.2 缓存多副本放置方法

在异构服务器组成的集群系统中, 最佳的负载均衡须根据服务器的性能选择需缓存的副本, 式(9)有所体现。本文将缓存服务器分为四类: 高性能高缓存服务器、高性能低缓存服务器、低性能高缓存服务器、低性能低缓存服务器。对于第一类和第四类服务器, 可以按照缓存对象的通信权重值按比例放置。而对于第二、三类服务器, 不仅要考虑其服务器的处理能力, 还需最大利用率地使用缓存空间。比如, 在高性能低缓存服务器中, 可以放置网络权重值高的副本, 提高热点瓦片的访问响应速度; 而在低性能高缓存的服务器中, 可以放置权重值较低的副本, 在不影响系统整体性能的条件下, 缓存更多的瓦片副本。 $w_i$  体现了瓦片  $t_i$  副本的通信开销。因此为达到负载均衡, 可根据  $w_i$  进行副本放置。

先将副本个数为  $N$  的瓦片进行放置, 每台服务器放置一个该瓦片的副本。放置完所有副本个数为  $N$  的瓦片之后, 则所有服务器当前的通信权重值都相等, 设为  $W'$ ; 且所有服务器当前已占用的缓存为  $C'$ 。根据副本生成策略, 剩下的副本权重值都相等, 为  $\bar{w}$ 。假设根据服务器的处理能力, 服务器  $S_i$  剩余的可缓存瓦片的空间为  $C'_i$ 。当前  $S_i$  缓存的副本访问权重值之和为  $W_i = \bar{w} \cdot C'_i + W'$ , 代入式(9), 有:

$$\frac{W' + \bar{w} \cdot C'_1}{V_1} = \frac{W' + \bar{w} \cdot C'_2}{V_2} = \dots = \frac{W' + \bar{w} \cdot C'_N}{V_N} = \bar{l} \tag{10}$$

假设  $V_1$  为性能最好且缓存容量较大的服务器, 用尽  $V_1$  剩余的缓存则能获取最好的资源利用率, 即  $C'_1 = C_1 - C'$ , 可计算得出  $\bar{l}$  值。代入  $\bar{l}$  可得:

$$C'_i = (\bar{l} \cdot V_i - W') / \bar{w} \tag{11}$$

根据式(11)可计算出服务器  $S_i$  的  $C'_i$ , 并检测服务器  $S_i$  的  $C'_i$  与  $C_i - C'$  之间的关系。

1) 若  $C'_i \leq C_i - C'$ , 则服务器属于高缓存低性能类型或高缓存高性能类型, 只需要放置  $C'_i$  个副本, 使其充分利用其处理能力, 避免服务器拥塞。

2) 若  $C'_i > C_i - C'$ , 则服务器属于低缓存高性能类型或低缓存低性能类型, 则需放置  $C_i - C'$  个副本即可, 充分利用其缓存。

该副本放置方法是基于式(9)和式(10)定义的平均负载值, 并利用副本生成方法产生的瓦片

副本具有相等的网络通信权重值特点,根据服务器的性能和其缓存空间大小,以高性能高缓存服务器为基准,计算与其处理能力匹配的缓存能力,进行副本的放置。如此,各异构的服务器根据其服务能力使其负载接近平均负载值,实现式(2)中负载不均衡度值最小,达到负载均衡最优的目标。同时,在避免服务器拥塞的同时充分利用其缓存,实现较高的资源利用率。

2.3 缓存多副本管理

为了更有效地、简洁地管理缓存多副本,及在分布式集群系统中高效地分配对瓦片及其副本的访问请求,本文在负载均衡器上建立缓存索引表  $Cached(t, St, Visited)$ 。其中,  $t$  为被缓存的瓦片对象,其值为瓦片的坐标三元组  $(tx, ty, l)$ ,并作为关键字自变量;  $St_{tx, ty, l}$  为缓存了瓦片  $t_{tx, ty, l}$  副本的服务器集合,  $St_{tx, ty, l} = \{St_{i, tx, ty, l} \mid 1 \leq i \leq r_{tx, ty, l}\}$ ,  $r_{tx, ty, l}$  为瓦片  $t_{tx, ty, l}$  的副本个数。由于瓦片副本的生成及其放置是基于负载均衡的,每个瓦片副本有着相同的网络通信权重值,因此负载均衡器可基于轮询方式转发瓦片的访问请求,该方法具有简洁性和高效性。使用  $Visited_{tx, ty, l}$  标识最后一次被访问的服务器  $St_{i, tx, ty, l}$ 。当负载均衡器收到对瓦片  $t_{tx, ty, l}$  的请求时,将请求转发给服务器  $St_{i+1, tx, ty, l}$ ; 若  $i = r_{tx, ty, l}$ , 则将请求转发给  $St_{1, tx, ty, l}$ 。  $Cached(t, St, Visited)$  是一个简单的线性链表,随着集群系统中缓存瓦片及其副本的更新,它根据键值  $(tx, ty, l)$  进行结点的增加、删除及其缓存信息的修改。因此,对于缓存多副本的管理,它是简洁的,其操作算法的时间复杂度与缓存瓦片数据的对象个数  $M$  相关,为  $O(M)$ 。

3 实验与分析

仿真环境中采用 9 个异构 Intel 双核 Linux 服务器,通过千兆以太网构建分布式集群系统,系统入口处放置一台有足够处理能力的负载均衡器。瓦片访问请求数为 100 000 个,以 Poisson 流到达。实验中模拟实现本文提出的最佳负载均衡的副本策略(OAL),并与经典的并仍广泛使用的副本策略 ORS 和 AMDR 进行比较。

分布式集群缓存容量是影响分布式集群缓存多副本策略效率的重要因素。采用同一层数据块大小为 256 像素×256 像素的瓦片进行分布式缓存多副本仿真。每台服务器的缓存能力大小不同,缓存能力从 1 000 个瓦片到 10 000 个瓦片不等,处理能力也不相同,但其能力值恒定。选取不

同个数的瓦片进行缓存副本对比实验,考察集群缓存容量变化时的系统性能指标,包括负载不均衡度、缓存命中率 and 请求平均响应延时。

图 2 为不同集群缓存容量下,三种多副本策略在异构的分布式集群缓存系统中的负载不均衡度对比,横坐标表示分布式集群缓存系统能提供的最大缓存空间。负载不均衡度依据式(2)计算得出。从图 2 可以看出,OAL 几乎达到负载均衡,相比 AMDR,其负载不均衡度低 8%左右;相比 ORS,其负载不均衡度低 9%~10%。从多副本策略上分析,OAL 基于两个负载平衡式(4)、式(9)生成副本及其部署,热度值较大的瓦片对象拥有  $N$  个副本,即每台服务器都拥有其副本;热度值较小的瓦片通信权重值相等,因此它更能实现负载均衡目标。

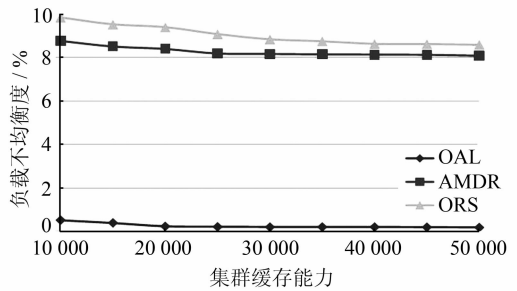


图 2 多副本策略在分布式集群缓存的负载不均衡度比较  
Fig. 2 Comparison of Load Imbalance Degree

图 3 为不同集群缓存容量下,三种多副本策略在异构的分布式集群缓存系统中的缓存命中率对比。可见,无论缓存瓦片多少,OAL 的命中率都高于其他两种策略。在低缓存空间下,OAL 相比 ORS 策略,其缓存命中率高达 14%左右;相比 AMDR 算法,其缓存命中率高达 8%左右。说明在缓存资源有限的条件下,OAL 更能匹配访问热点的分布,使缓存资源得到有效的利用。当分布式集群缓存系统的缓存容量达到 25 000~30 000 个瓦片能力时,集群系统的缓存命中率可以达到平衡。这说明在集群服务器处理能力一定的条件下,增加缓存容量意义不大;同时也验证了瓦片的访问具有重尾特征,即少数瓦片吸引多数访问。实验说明访问热度值大的瓦片对象可赋予较多的副本个数,而热度值小的瓦片对副本策略的性能影响不大。

分布式集群系统的平均请求响应时间除了与网络带宽、系统服务能力有关,还与用户的请求并发数紧密相关,即与用户的访问密度相关。根据图 1 的网络模型,用户的访问密度可用单位时间内客户端的请求个数与集群系统服务器处理能力

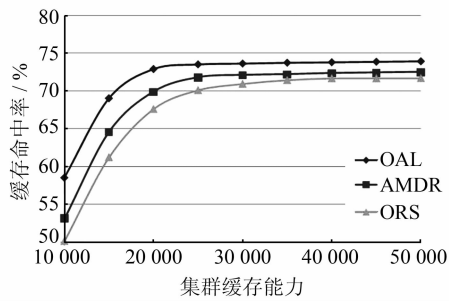


图 3 多副本策略在分布式集群缓存的缓存命中率比较  
Fig. 3 Comparison of Cache Hit Rate

的比值标识,其比值越大,表明当前用户的访问越密集。其中,集群系统的服务器处理能力是指单位时间内服务器能处理的请求个数,其值为  $\sum_{i=1}^N V_i/\mu$ 。本文实验中使用的用户访问密度值为 0.6,测试网络 GIS 在偏重负荷下的平均响应时间。

图 4 为不同集群缓存容量下,三种多副本策略在异构分布式集群缓存系统中的平均请求响应时间对比。从图 4 可以看出,随着分布式集群系统缓存空间的增大,各个策略的请求响应时间逐渐减小,在缓存空间为 35 000 时逐渐达到平衡。但是在任何缓存条件下,OAL 比其他两种算法的平均响应都快,且其平均响应时间下降的幅度较为平缓。在低集群缓存条件下,OAL 的请求响应时间比 AMDR 算法减少 170~130 ms 左右,比 ORS 减少 170~210 ms 左右。在集群缓存空间为 25 000~50 000 时,OAL 比 AMDR 算法的响应时间减少 90~100 ms,比 ORS 减少 125 ms 左右。实验结果说明,在服务器能力异构的集群系统中,具有较小负载不均衡度的副本策略能较大地提高集群服务的整体性能,增强用户的访问体验。特别是在有限的缓存条件下,比其他算法更能适应用户的密集访问。

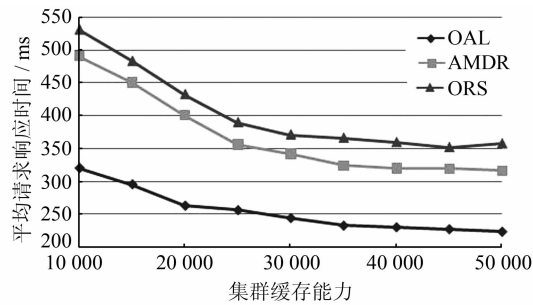


图 4 多副本策略在分布式集群缓存的平均请求响应时间比较  
Fig. 4 Comparison of Average Request Response

4 结 语

分布式缓存副本策略是网络地理信息服务中实现大规模高效并行服务的方法之一,而异构服务器集群是目前云计算中的网络地理信息服务常见的环境。本文以负载均衡为目标,针对网络地理信息服务中大规模用户访问的不均衡性和异构服务器性能的不均衡性,最大化利用缓存资源的同时,匹配服务器处理性能生成副本及部署副本。该策略实现了异构分布式缓存集群系统的整体性能优化,能适应大规模的用户密集访问。未来工作可进一步考虑如何定量定性地分析服务器的缓存能力和处理能力在集群环境中的关系比,加入适应空间数据访问热点变化的缓存多副本置换机制,实现快速的副本置换,将获取更好的实际应用效果。

参 考 文 献

[1] Wu Huayi, Zhang Hanwu. QoGIS: Concept and Research Framework[J]. *Geomatics and Information Science of Wuhan University*, 2007, 32 (5): 385-388(吴华意, 章汉武. 地理信息服务质量(QoGIS):概念和研究框架[J], 武汉大学学报·信息科学版, 2007, 32 (5): 385-388)

[2] Lou Shurong, Meng Lingkui, Fang Jun, et al. The Network Transmission Model of Multiresolution Image Based on Peer-to-Peer [J]. *Acta Geodaetica et Cartographica Sinica*, 2011, 40(5): 628-634(娄书荣, 孟令奎, 方军, 等. 基于对等网络的多分辨率影像的网络传输模型[J]. 测绘学报, 2011, 40(5): 628-634)

[3] Serpanos D N. MMPacking: A Load and Storage Balancing Algorithm for Distributed Multimedia Servers[J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 1998, 8(1): 13-17

[4] Dukes J, Jones J. Dynamic Repacking: A Content Replication Policy for Clustered Multimedia Servers [R]. Trinity College Dublin Computer Science Department Technical Reports, Cambridge, UK, 2002

[5] Ibarkai T, Katoh N. Resource Allocation Problem—Algorithmic Approaches[M]. Massachusetts United States: The MIT Press, 1988

[6] Zhou Xiaobo, Xu Chenzhong. Optimal Video Replication and Placement on a Cluster of Video-on-Demand Servers[C]. The 2002 International Conference on Parallel Processing, IEEE Computer Society, Washington D C, USA, 2002

[7] Zhuo Juchao. Study of Cache Placement for Time-

shifted TV Cluster Using Genetic Algorithm[D]. Hefei:University of Science and Technology of China,2010 (卓居超. 时移电视集群系统缓存调度研究[D]. 合肥:中国科学技术大学,2010)

[8] Dan A, Sitaram D. An Online Video Placement Policy Based on Bandwidth to Space Ratio(BSR)[C]. ACM SIGMOD’95, New York , USA,1995

[9] Fu Wei, Xiao Yi, Lu Xicheng. QoS Perception Placed a Copy of the Studies Reviewed[J]. *Research and Development of the Computer* , 2009, 46 (2): 36-43 (付伟,肖依,卢锡城. QoS 感知的副本放置问题研究综述[J]. 计算机研究与发展,2009,46(2): 36-43)

[10] Fisher D. Hotmap: Looking at Geographic Attention[J]. *IEEE Transactions on Visualization and Computer Graphics* ,2007,13(6):1 184-1 191

[11] Wang Hao, Pan Shaoming, Peng Min, et al. Zipf-like Distribution and Its Application Analysis for Image Data Tile Request in Digital Earth[J]. *Geomatics and Information Science of Wuhan University* ,2010,35(3):356-359(王浩,潘少明,彭敏,等. 数字地球中影像数据的 Zipf-like 访问分布及应用分析[J]. 武汉大学学报·信息科学版,2010,35(3): 356-359)

[12] Li Rui, Zhang Yinfeng, Xu Zhengquan, et al. A Load-balancing Method for Network GISs in a Heterogeneous Cluster-based System Using Access Density[J]. *Future Generation Computer Systems* , 2013,29(2):528-535

## A Replication Strategy Based on Optimal Load Balancing for a Heterogeneous Distributed Caching System in Networked GISs

LI Rui<sup>1,2</sup>   TANG Xu<sup>3</sup>   SHI Xiaolong<sup>1</sup>   FAN Jiapei<sup>1</sup>   GUI Zhipeng<sup>2,4</sup>

1   State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China

2   Collaborative Innovation Center of Geospatial Technology, Wuhan 430079, China

3   School of Resource and Environmental Sciences, Wuhan University, Wuhan 430079, China

4   School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079 , China

**Abstract:** Networked geospatial information services in cloud-based environments are distributed and heterogeneous; accesses to geospatial data (tiles) are uneven and has the feature of a high degree of aggregation. Aiming to minimize the degree of load imbalance, this paper proposes a replication strategy for a heterogeneous distributed high-speed cluster-based caching system. First, taking into account the unbalanced accesses to tiles, it minimizes the weighted communication values of hotspot tiles, and generates the maximum number of replicas based on the total cache capability of the distributed cluster-based caching system. Then, since each server has a different processing capacity in the heterogeneous system, the strategy places the replicas based on the service performance of each caching server and the weighted communication value of each replica, thus matching the cache capacity of each server. Experimental results reveal that the proposed strategy can avoid server congestion while fully utilizing limited cache capacity to achieve a better load balancing and a high resource utilization, delivering good response performance and a high cache hit rate.

**Key words:** load balancing; cluster; heterogeneous; cache; QoS

**First author:** LI Rui, PhD, associate professor, specializes in spatial cloud computing, networked GIS. E-mail: ruili@whu.edu.cn

**Corresponding author:** GUI Zhipeng, PhD, lecturer. E-mail: zhipeng.gui@whu.edu.cn

**Foundation support:** The National Natural Science Foundation of China, No. 41371370; the National High Technology Research and Development Program (863 Program) of China, No. 2012AA12Z401.