

面向流媒体基于蚁群的副本选择算法^①

杨 戈, 刘智鹏

(北京大学深圳研究生院, 深圳物联网智能感知技术工程实验室, 深圳 518055)

通讯作者: 杨 戈, E-mail: yangge@pkusz.edu.cn



摘 要: 本文针对流媒体 Cloud-P2P 存储模式中的副本选择, 提出一种基于蚁群算法的改进算法 (replica selection algorithm in Cloud-P2P based on ant colony algorithm, C²P²RSA²), 建立副本选择度量标准 (副本节点的网络带宽、网络延时等) 与蚁群信息素的映射, 定义了副本信息素概率, 最后得到一组副本资源的最优解. 实验表明, 与 PARS A 算法 (Pheromone-base Ant colony Replica adaptive Selection Algorithm in cloud storage) 和最佳副本选择算法比较, 在平均访问时间增加 2%~5% 的情况下, 本文的算法对云副本节点的负载率减少 15%~25%.

关键词: Cloud-P2P; 蚁群算法; 副本选择

引用格式: 杨戈, 刘智鹏. 面向流媒体基于蚁群的副本选择算法. 计算机系统应用, 2019, 28(2): 190-195. <http://www.c-s-a.org.cn/1003-3254/6590.html>

Replica Selection Algorithm Based on Ant Colony Algorithm for Streaming Media

YANG Ge, LIU Zhi-Peng

(Engineering Lab on Intelligent Perception for Internet of Things (ELIP), Shenzhen Graduate School, Peking University, Shenzhen 518055, China)

Abstract: Replica selection algorithm for streaming media in Cloud-P2P (C²P²RSA²) is proposed based on ant colony algorithm in this study. In replica selection model based on ant colony, a copy select metrics (copy node network bandwidth, network delay, etc.) is mapped by ant colony pheromone. The replica pheromone probability formula is proposed. Through constantly iterate the optimal replica resource is selected. The experimental results show the average access time of C²P²RSA² is increased by 2%~5% than that of PARS A (pheromone-base ant colony replica adaptive selection algorithm in cloud storage) and best copy selection algorithm, the cloud replica node load factor of C²P²RSA² is decreased by 15%~25%.

Key words: Cloud-P2P; ant colony optimization; replica selection

1 引言

流媒体网站的注册用户自 2008 年起连续不断地增加, 从 2009 年到 2017 年, 中国的流媒体用户数量一直以 15%~20% 的速率不断增加, 固然在用户规模的增

加速率在近几年有所放缓, 但是用户规模依然平稳上升.

如今的网络上网络传输的方式有很多种, 但是多半是以 C/S 模式还有 P2P 模式, C/S 模式是由两个机

① 基金项目: 国家自然科学基金 (61272364); 广东高校省级重大科研项目 (201612008QX); 广东省学科建设专项资金 (2013WYXM0122); 广东省大学生创新创业训练计划项目 (201613177024, 201713177036); 深圳市科技计划项目 (JCYJ20160428153620486, JCYJ20170303140803747); 深圳市智能媒体和语音重点实验室资助项目 (201762005)

Foundation item: National Natural Science Foundation of China (61272364); Provincial Major Science and Technology Research Program of Higher Educations of Guangdong Province (201612008QX); Special Fund for Discipline Construction of Guangdong Province (2013WYXM0122); Innovation Training Program for Students of Higher Education of Guangdong Province (201613177024, 201713177036); Science and Technology Program of Shenzhen Municipality (JCYJ20160428153620486, JCYJ20170303140803747); Open Fund of Shenzhen Laboratory of Intelligent Media (201762005)

收稿时间: 2018-03-03; 修改时间: 2018-03-19, 2018-04-02; 采用时间: 2018-04-12; csa 在线出版时间: 2019-01-28

器组成的,一个为服务器,另外一个为用户端.他们之间相互发送数据,C/S模式有一个致命的缺点就是要求服务器的集群性能非常高,这个缺点随着用户基数的增加,越明显的表现出来.

P2P模式是可以直接互换来获取本身所须要的资源,在对等网络里每一台主机都是同等的,都可以信息共享,即充任用户机也可以充任服务器.较原来所有的C/S模型效率提升很多,提高了可靠性与可用性,但是如果在用户数量不多的前提下,P2P模式下观看的流媒体,不会因为传输速率而影响到了用户的观看的感受.但是如果人数迅速增加或者在达到一定数量时,其服务质量则难以保证.在P2P模式中,用户数量不断增长,流媒体的系统会变得难以管理.

本文将引入一种新的储存方式Cloud-P2P来解决流媒体文件的储存问题,Cloud-P2P的储存结构是一种比较新的储存方式,它将与传统传输模式P2P,和现代较新的技术云计算结合起来,将云计算上的优点与P2P上的优点相结合.Cloud-P2P储存方式是通过云服务中心和P2P储存节点组成的.

由于在云环境下的流媒体的副本数量很大,且不同的副本选择,会导致副本所在虚拟服务器的负载不同,一个号的副本选择策略会使虚拟服务器的负载达到一个比较理想的值,所以本文提出基于蚁群算法的Cloud-P2P副本选择算法($C^2P^2RSA^2$),来对副本进行选择.

2 相关工作

2.1 Cloud-P2P

文献[1]中Cloud-P2P储存结构是由云服务中心和P2P储存节点组成的,云服务中心利用流媒体的提供商的众多服务器组成,利用多个物理服务器,通过云计算平台整合成一个大的资源池,在从资源池里虚拟出多个虚拟服务器,在每个不同时间里,用户的请求量是不同的,在一些不繁忙的时间段,虚拟服务器可以从资源池里获取少量的资源,就能够满足响应用户的请求.在一些繁忙时间段里,有大量的用户请求,虚拟服务器可以从资源池里调度大量的资源,使得短时间内虚拟服务器的处理能力大大的提升,快速响应用户的请求.虚拟服务器不仅可以为用户提供服务,而且还能管理P2P里的储存节点,查看节点的在线情况以及监控节点的负载等等的信息.

用户端的程序可以充当P2P节点,多个P2P节点

就能组成一个很大的储存空间,如果服务器发生了故障,在储存空间上的数据也能维持给用户使用,并且储存节点上的可以修复服务器.

现在的云存储系统结构主要是(Master/Slave)结构的GFS、HDFS,文献[2]提出一种对等云结构的云存储系统Ming Cloud,并且通过构建原型系统,使Ming Cloud能实现数据的存储、读取、删除、搜索等功能.

2.2 副本选择算法

在流媒体系统中,要想用户获得较好的观看质量,系统会复制多份副本出来,在众多的副本中,哪一个副本适合用户,使用户获得较好的观看质量,这个就是副本选择的问题,在某些情况下并非副本响应时间最短就是最佳的副本节点,还要看用户使用的需求,所以提出QoS偏好感知的副本选择策略,即QoS的优化.还有一些仿生的算法如粒子群算法、遗传算法等等.

文献[3]提出了在某些情况下并非副本响应时间最短就是最佳的副本节点,还要看用户使用的需求,所以提出QoS偏好感知的副本选择策略,即QoS的优化.

文献[4]资源的合理配置能有效地提高非结构化P2P网络的查询性能,提高资源的可用性.目前的研究报告的副本,资源多集中在各类定量分析的副本和分布式资源分配策略的配置,节点独立地选择资源配置,没有考虑P2P网络节点配置的节点之间的相互作用仅保持连接数和邻居节点,掌握该本地信息,所以在交互过程可视为有限理性节点.在查询性能分析和资源分配节点基于结构查询节点性能相关的收益函数的关系节点,资源分配问题转化为一个进化的博弈,可以在资源配置过程中的节点和查询性能的有效互动分析可以通过进化的方法得到说明.仿实验结果表明,资源配置模式可以获得更高的查询成功率和最优跳数的平均数,并保持相对较低的冗余.

文献[5]提出根据数据节点和网络链路的可靠性因子分析,云存储系统的一个数据副本服务可靠性模型.根据访问的可靠性和数据之间的副本数量,用户访问,设计的数据业务的可靠性,复制产生时间和存储节点选择方法来确定副本分布和删除算法,并在云存储系统的ERS-Cloud进行了一系列的实验.结果表明,可以有效地保护数据服务的可靠性的方法,进一步降低冗余存储的份数.

文献[6]提出了在无线网络,由于无线网络的不稳定性,导致流媒体的传输受到影响,所以有一种基于跨

层码率适配和差错控制的流媒体无线传输方法. 文献[7]提出了随着云计算的普及使用, 很多的数据都使用云存储技术, 将数据或者副本上传到云上, 所以如何在云存储环境下进行对副本的选择也成为了人们研究的一个方向, 通过优化后的粒子群算法, 对副本进行选择.

2.3 蚁群算法

在1992年的时候, Dorigo 和 Maniezzo 第一次提出了蚁群算法 (Ant Colony Optimization, ACO). 在1997年的时候, Dorigo 提出了精英策略的蚁群算法 (Ant System with elitist strategy, ASelitist). 在1997年的时候 X Bullheimer B 和 Hartl RF 提出了基于排序的蚁群算法 (Ant System based in rank, ASrank). 在2000年的时候, Stutzle T 和 Hoos HH 提出了最小最大蚁群算法 (Max-Min Ant System, MMAS).

文献[8]提出从仿生角度出发提出一种全新的网络聚类算法——基于随机游走的蚁群算法 RWACO. 该算法将蚁群算法的框架作为 RWACO 的基本框架, 对每一代, 马尔可夫随机游走模型作为启发式规则的基础上, 集成学习的思想, 蚂蚁局部解决方案融入全局解决方案, 和更新的信息素矩阵. 通过加强内部集群连接, 削弱了跨集群连接“这一进化策略, 逐渐显现集群网络结构, 实验结果表明, 对一些典型的计算机生成网络和真实网络, 该算法能够准确地探索网络来衡量的真实数量集群, 与一些有代表性的算法具有较高的聚类准确率比^[9,10].

如今很多的数据副本等都放置在云上, 所以在云上如何对副本你的选择也成为了一个问题, 文献[11]提出了一种基于云储存的副本选择算法 (Pheromone-base Ant colony Replica adaptive Selection Algorithm in cloud storage, PARS). PARS 算法是对存储在云上的副本进行选择的, 如果使用此算法在 Cloud-P2P 的储存上, 由于云副本节点由于处于一个比较好的网络环境下, 这样会造成云副本节点会被用户大量的使用, 这样会对服务提供商增加运营成本^[12], 所以本文算法对此进行优化修改, 使优化后的算法可以在 Cloud-P2P 下使用, 优化过后副本节点会优先使用 P2P 上的副本节点, 在 P2P 的副本节点的负载率到达一定程度时, 云副本节点会被用户开始使用, 使用户获得正常的观看质量.

3 C²P²RSA² 算法

为了提高用户观看流媒体的时候, 有流畅的观看

感受和良好的观看质量, 服务提供商会增加副本的数量, 给用户使用. 但在众多的副本中, 找到适合用户所需要的副本. 蚁群算法使一种适合分布式系统的算法, 而且蚁群算法拥有正反馈, 系统可以根据反馈不停改变系统的副本, 使用户获得更好的观看感受. 本文算法优化了蚁群算法, 会优先选择 P2P 上的副本节点, 减少云副本节点的使用, 使服务商的成本降低.

(1) 利用初始化公式对其初始化, 影响副本的选择的参数有很多, 例如副本读取的时间、网络的带宽、网络延迟、副本节点的负载完成率 and 是否为云副本节点等等, 通过这些参数利用初始化公式使每个副本都有各自的初始信息素数值.

信息素数值: 初始的信息素数值是通过副本节点的网络环境进行初始的, 并且通过副本节点网络环境和处理能力进行动态调整.

$Td_a^j(e)$: 表示用户 a 到副本 j 的路径 e 的副本节点的副本读取时间, 某一时刻 d 的副本读取时间, 记为 w_d , 最大副本的读取时间为 w_{max} .

$delay_a^j(e)$: 表示用户 a 到副本 j 的路径 e 的网络延迟, 本文中的网络延迟是指网络跳数, 最大网络延迟为 m_{max} , 最小网络延迟为 m_{min} .

$bw_a^j(e)$: 表示用户 a 到副本 j 的路径 e 的网络带宽, 某时刻 d 副本路径的网络带宽记为 n_d .

$lr_a^j(e)$: 表示用户 a 到副本 j 的路径 e 上副本节点的负载完成率, 某时刻 d 副本节点的已处理完的副本数记为 y_d , 副本节点总的处理任务数为 y_0 .

$$\tau(0)_j = \frac{Td_a^j(e)}{f_size} + \gamma_j \quad (1)$$

式 (1) 是对副本 j 进行信息素的初始化, f_size 为副本文件的大小, $Td_a^j(e)$ 为用户 a 到副本 j 的读取时间, γ_j 为初始化信息素数值.

式 (2) 中为副本 j 的初始的总信息素数值, 由五个参数组成的, 即:

$$\begin{aligned} \gamma_j = & \varepsilon_{Td_a^j(e)}^d * 100\% + \varepsilon_{delay_a^j(e)}^d * 100\% + \varepsilon_{bw_a^j(e)}^d * 100\% \\ & + \varepsilon_{lr_a^j(e)}^d * 100\% + iscloud * 100\% \end{aligned} \quad (2)$$

式 (2) 中的几个参数分别由下列的式 (3)、式 (4)、式 (5)、式 (6) 中进行初始化得到的.

$$\varepsilon_{Td_a^j(e)}^d = (w_{max} - w_d) / w_{max} \quad (3)$$

$$\varepsilon_{delay_a^j(e)}^d = (m_{\max} - m_d) / m_{\min} \quad (4)$$

$$\varepsilon_{bw_a^j(e)}^d = (n_{\max} - n_d) / m_{\max} \quad (5)$$

$$\varepsilon_{r_a^j(e)}^d = y_d / y_0 \quad (6)$$

iscloud: 表示是否为云副本节点, 如果 P2P 副本节点的负载率到达一个指定的值, 此参数会将云副本节点的信息素浓度增高, 使用户可以使用云副本节点, 不会使用户观看的质量下降。

(2) 利用式 (7) 计算副本选择概率:

$$p_j^k = \begin{cases} \frac{(\tau_j(t))^\alpha \cdot (\vartheta_j)^\beta}{\sum_{j=1}^q (\tau_j(t))^\alpha \cdot (\vartheta_j)^\beta}, & j, q \in Z^+ \\ 0, & \text{else} \end{cases} \quad (7)$$

$\tau_j(t)$: 表示在 t 时刻, 副本 j 的信息素浓度。

ϑ_j : 表示副本 j 初始的信息素数值。

α 、 β : 分别表示在 t 时刻, 信息素浓度的重要性以及副本初始信息素数值的重要性。

(3) 信息素更新, 式 (8) 是副本被成功选择或者选择失败是, 信息素都会改变的, 这是就会通过该式进行修改. 式 (9) 是当副本被选出来的时候, 对副本的信息素浓度进行更新

$$\tau_j(t+1) = \rho * \tau_j(t) + \tau_j \quad (8)$$

$$\tau_j(t+1) = \rho * \tau_j(t) + (1 - \rho) \cdot \varphi * \tau_j \quad (9)$$

τ_j : 表示为信息素的改变量;

ρ : 表示信息素的残留系数;

φ : 表示为扩大系数, 如果有适合的副本的时候, 该系数可以使信息素快速的增加。

图 1 中为算法的流程图, 下面是算法的流程:

(1) 对每个副本节点进行第一次的信息素浓度初始化, 可以依据初始化信息素公式。

(2) 依据概率公式对每个副本节点的选择概率进行计算。

(3) 利用概率匹配的原则, 选出副本节点。

(4) 经过一轮选择以后, 一些被蚂蚁选出来的副本, 利用奖励公式将其的副本信息素浓度进行更新, 一些没有被选择到的副本节点, 则使用惩罚公式将其副本信息素浓度进行更新。

(5) 依据终止条件, 如果符合了条件, 就输出副本选择的结果, 如果不满足的, 则重新回到步骤 (2) 继续

下一轮的迭代。

4 实验结果及分析

本次实验所使用的仿真环境如表 1, 利用 Visual Studio 2012 Ultimate 对仿真环境的生成, 并且对实验中要生成的多个副本节点中的副本读取的时间、网络延迟、网络带宽都是随机产生的, 随机数服从均匀分布, 均匀分布是指在范围内的数字出现的机率相等。

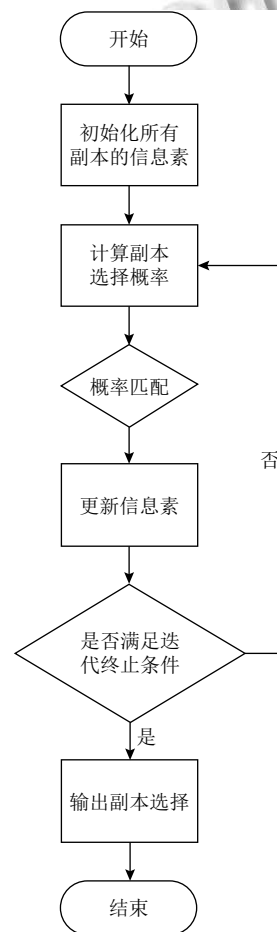


图 1 C²P²RSA² 算法流程

表 1 仿真实验的软硬件参数

| 设备 | 处理器 | 内存 | 操作系统位数 | 硬盘 | 编程软件 | 操作系统 |
|----|----------|---------|--------|----|--------|----------|
| 参数 | 1.70 GHz | 8.00 GB | 64 位 | 1T | VS2012 | Windows7 |

副本读取时间的取值范围: 10–120 M/s; 网络延迟的范围 (网络跳数的范围): 1–255; 网络带宽的范围: 1–100 M。

利用以上随机生成的数据生成出副本节点来, 可以模拟出现实环境中的不同网络环境的副本节点^[1]。

本实验从选择 Cloud-P2P 网络中副本节点的平均访问时间进行了分析, 通过与最佳副本选择算法^[1]与 PARS A 算法进行比较, 得出了结论, 证明了本文的 C²P²RS A² 算法的可行性和优点, 本文利用平均访问时间、云副本节点的负载率作为评价的标准, 每个实验重复做 3 次, 取平均值。

平均访问时间: 是指每个用户从提出请求开始, 副本节点响应, 传输给用户, 并且用户完全接受后, 总共的时长。

云副本节点的负载率: 云副本节点上的平均资源的使用率。

图 2 为当副本节点有 10 个的时候, 本文算法与 PARS A 算法和最佳副本选择算法的平均访问时间的对比, 由图上可以看到当副本节点少的时候, 三种算法的平均访问时间都差不多, 但总体本文算法比最佳副本选择算法少, 但是比 PARS A 算法多 2.1%。

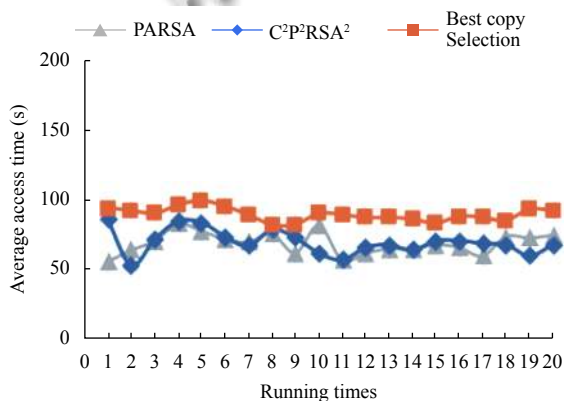


图 2 副本节点为 10 的平均访问时间对比图

图 3 为当副本节点的数量为 20 的时候, 本文算法与 PARS A 算法和最佳副本选择算法的平均访问时间的对比, 由图上可以看到当副本节点增加了后, 最佳副本选择算法与本文算法和 PARS A 算法的平均访问时间差别开始增大, 本文算法比最佳副本选择算法少, 但是比 PARS A 算法多 4.3%。

图 4 为副本的平均访问时间的对比, 刚开始由于副本数量不多, 所以平均访问时间都差不多, 随着副本的数量增加, 差距开始增大, 由图上可以看到最佳副本选择算法比本文算法 C²P²RS A² 与 PARS A 的平均访问时间都要多, C²P²RS A² 与 PARS A 相比, 前者比后者的平均访问时间增加 2%~5%。

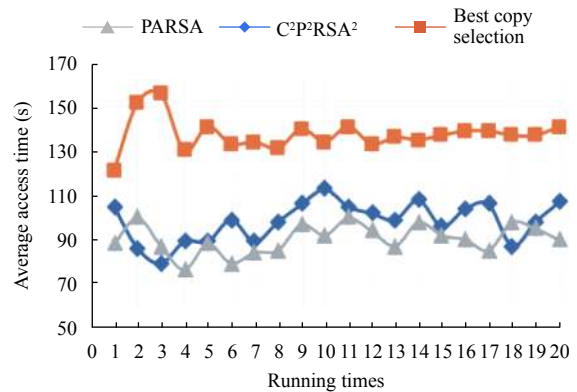


图 3 副本节点为 20 的平均访问时间对比图

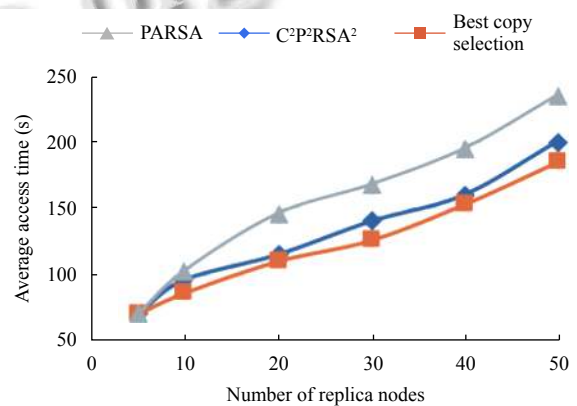


图 4 副本平均访问时间对比

图 5 为本文的 C²P²RS A² 算法与 PARS A、最佳副本选择算法的云副本节点的负载率对照图, 本文设置的云副本节点为 5 个节点, 较少的云副本节点当用户选择的副本不是云副本节点的时候, 可以从云副本平均负载率快速反映出来, 固定云副本节点数量, 可以通过改变 P2P 副本节点的数量, 可以观察出不同的算法的云副本负载率不一样, 刚开始由于没有 P2P 的副本节点, 所以所有的副本都有云副本节点来提供, 随着 P2P 上的副本节点增加, 因为最佳副本选择算法没有动态调整的机制, 所以云副本负载率没有降低, 而 PARS A 算法通过动态调整, 使云副本节点的负载率有微小降低, 但由于云副本节点的网络环境方面都很好, 所以在用户选择副本的时候, 还是会对云副本节点进行优先选择, 而本文的 C²P²RS A² 有动态调整机制, 并且会对 P2P 副本节点进行优先选择, 所以当副本节点开始增加的时候, 云副本节点的负载率降低, 并且随着 P2P 副本节点的数量增加而逐渐下降。

通过图 2 至图 5 上的数据可知, 本文 C²P²RS A² 算

法的平均访问时间小于最佳副本选择算法,最佳副本选择算法也是通过结合了网络延迟、网络带宽、副本节点的可靠性等计算出最小代价副本,而本文的 $C^2P^2RSA^2$ 算法则是根据副本节点的副本读取时间、网络延迟、网络带宽、服务器的负载率等因素综合起来,而且不断动态调整,然后对副本进行选择的,所以通过实验证明了本文的 $C^2P^2RSA^2$ 算法优于最佳副本选择算法,本文的 $C^2P^2RSA^2$ 算法能在平均访问时间与PARSA算法上的时间相差2%~5%,但是在负载率上可以看到,本文的 $C^2P^2RSA^2$ 算法能降低云副本的负载率15%~25%,提高P2P副本节点的负载率,优先对P2P副本节点进行使用,降低云副本节点的负载率。

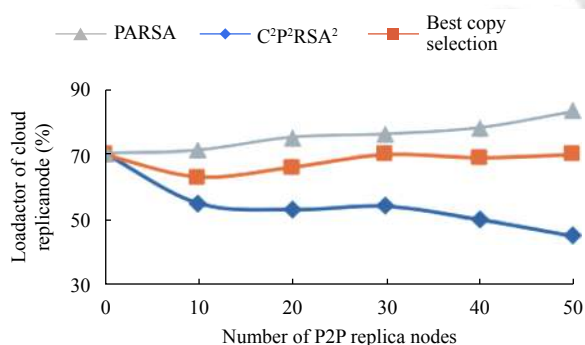


图5 云副本节点的负载率对比

5 总结

为了能满足人们可以观看流媒体,本文使用的Cloud-P2P的存储模式是通过云中心与P2P节点进行结合,使用云的灵活性、按需付费的特性与P2P的可扩展性相结合。为了能对Cloud-P2P的存储模式进行副本选择,本文提出了基于蚁群算法的 $C^2P^2RSA^2$ 。客户使用本算法进行对副本选择的时候,优先使用P2P副本节点,对云副本节点的负载率降低,在云副本节点的负载率降低到一定程度可以减少云副本的节点,在P2P副本节点不能使新的用户正常使用的时候,可以增加云副本节点使用户的观看质量不会下降,通过云计算的灵活性可以做到这一点。在后续的工作中,由

于蚁群算法要经过多次迭代才能获得最优的结果,这样计算速度很慢,下一步工作可以对迭代次数进行优化,减少迭代次数,能快速获得最优结果,并且在云副本节点的负载率降低到一定程度时,对云副本节点进行选择性的减少,减少云上的开销。

参考文献

- 金顺福,王晨飞,陈玲玲,等. Cloud-P2P云存储结构的模型建立与性能分析. 通信学报, 2015, 36(3): 2015077-1-2015077-8.
- 吴吉义,傅建庆,平玲娣,等. 一种对等结构的云存储系统研究. 电子学报, 2011, 39(5): 1100-1107.
- Zhao P, Huang TL, Liu CX, *et al.* Research of P2P architecture based on cloud computing. 2010 International Conference on Intelligent Computing and Integrated Systems. Guilin, China. 2010. 652-655.
- Huang LC. Semantic P2P networks: future architecture of cloud computing. 2011 Second International Conference on Networking and Distributed Computing. Beijing, China. 2011. 336-339.
- 熊润群,罗军舟,宋爱波,等. 云计算环境下QoS偏好感知的副本选择策略. 通信学报, 2011, 32(7): 93-102. [doi: 10.3969/j.issn.1000-436X.2011.07.011]
- 刘金霞,刘延伟,慈松. 基于跨层码率适配和差错控制的3D视频无线传输方法. 电子学报, 2014, 42(2): 312-318. [doi: 10.3969/j.issn.0372-2112.2014.02.016]
- 张翠苹,郭振洲,拱长青. 云存储环境下副本选择策略研究. 计算机科学, 2015, 42(S2): 408-412.
- 周经亚,宋爱波,罗军舟. P2P网络中一种基于进化博弈的资源配置模型. 软件学报, 2013, 24(3): 526-539.
- 金弟,杨博,刘杰,等. 复杂网络簇结构探测——基于随机游走的蚁群算法. 软件学报, 2012, 23(3): 451-464.
- 左方,何欣. 一种基于蚁群算法的云存储副本动态选择机制研究. 计算机应用研究, 2015, 32(11): 3368-3370, 3374. [doi: 10.3969/j.issn.1001-3695.2015.11.039]
- 黄昌勤,李源,吴洪艳,等. 云存储系统中数据副本服务的可靠性保障研究. 通信学报, 2014, 35(10): 89-97. [doi: 10.3969/j.issn.1000-436x.2014.10.011]
- 柴洁. 数据网格副本选择策略的研究[硕士学位论文]. 武汉: 武汉理工大学, 2007.