

基于增强学习算法的云资源动态弹性伸缩

张继东, 曹靖城, 周 帅

(天翼智慧家庭科技有限公司, 南京 210001)

摘要: 为解决云资源节点负载不均衡导致利用效率低、资源负载均衡性较差等问题, 提出以增强学习算法改善云资源动态负载均衡弹性伸缩的方法。采用云资源动态特性基准分析云资源动态的最初基准, 对云资源动态特性进行节点采样, 设置云资源节点负载的上限和下限, 实现云资源动态负载均衡控制; 采用非均匀离散傅里叶变换方法, 对其节点负载状态均匀调整, 引入增强学习算法中 Q 学习方法, 不断自适应执行弹性收缩策略, 实现云资源动态负载均衡的智能弹性伸缩。实验结果表明, 采用所提方法后, 云资源利用效率高达约 97%, 资源负载均衡性能得到提升。

关键词: 增强学习算法; 云资源; 负载均衡; 弹性收缩; 动态基准

中图分类号: TP391 **文献标识码:** A **文章编号:** 1009-2552(2021)08-0122-05

DOI: 10.13274/j.cnki.hdzj.2021.08.023

Dynamic elastic scaling of cloud resources based on reinforcement learning algorithm

ZHANG Ji-dong, CAO Jing-cheng, ZHOU Shuai

(Tianyi Smart Home Technology Co., Ltd., Nanjing 210001, China)

Abstract: In order to solve the problems of low utilization efficiency and poor resource load balance caused by unbalanced load on cloud resource nodes, a study is proposed to improve the elastic scaling of cloud resource dynamic load balance to enhance learning algorithm. The dynamic characteristics of cloud resources is adopted to analyze the dynamic characteristics of cloud resources. Node sampling is conducted on the dynamic characteristics of cloud resources, and upper and lower limits of cloud resource node load are set to realize the dynamic load balancing control of cloud resources. The non-uniform discrete Fourier transform method is adopted to adjust the load state of the nodes uniformly, the Q learning method in the reinforcement learning algorithm is introduced, and the elastic shrinkage strategy is continuously adaptive to realize the intelligent elastic shrinkage of cloud resources with dynamic load balancing. The experimental results show that the cloud resource utilization efficiency is up to about 97% after adopting the proposed method, and the resource load balancing performance is improved.

Key words: enhancement learning algorithm; cloud resources; load balancing; elastic contraction; dynamic benchmark

0 引言

随着电子信息技术不断发展, 云平台不断被应用到多个领域^[1], 其借助互联网为不同层次使用者提供云资源。云资源作为一种按需分配资

作者简介: 张继东(1976-), 男, 博士, 高级工程师, 研究方向为计算机信息技术、电信技术。

源, 其典型特点为资源动态伸缩性^[2], 为使用者动态扩展资源过程即为弹性伸缩。在互联网日益复杂环境中, 云资源弹性调度也出现在线迁移困难等问题^[3]。因此, 提高云资源弹性收缩性能成为该领域研究焦点。

文献[4]分析 Docker 容器数据卷中存在的问

题,将 Docker 和 Ceph 的特点相融合,构建云资源调度的优化模型。该方法对云资源粒度分层,实现云平台资源弹性调度,但资源调度运营成本高。文献[5]设置云资源动态调度限制条件,构建弹性资源的动态模型,引入遗传算法设计弹性云资源动态调度。该方法云资源调度稳定性好,但资源负载均衡性差。

为此,本文提出以增强学习算法提高云资源动态负载均衡的弹性收缩方法。通过对云资源动态特性分析,以及动态负载均衡控制,引入 Q 学习方法对云资源不断进行迭代,实现云资源动态负载均衡弹性伸缩。

1 云资源动态负载均衡控制

为了实现云资源的智能弹性收缩,首先需要实现云资源动态负载均衡控制^[6]。通过分析云资源动态特性和动态负载特性的采样,实现云资源动态负载均衡控制。

1.1 云资源动态特性分析

在云资源中存在较多相似且重复的数据,用户在选择所需云资源时难以区分,导致资源利用效率较低^[7]。因此,通过分析云资源的动态特性,提高云资源的利用效率。相似云资源虽然存在一定相近度,但其动态特性存在明显差异。本文通过引入云资源动态特性的基准^[8]对动态特性进行分析。云资源动态特性的固定基准是区别同一来源相似云资源的基准。

首先对云资源动态特性来源进行分析,获取其区分的原始基准。假设 S_i 表示云资源来源的最初标准, S_i^q 表示云资源来源 q 渠道中各云资源动态特性, T_q 代表云资源数据的总体数量, φ_q 代表云资源来源的相似度,则云资源动态特性的最初基准为:

$$S_i = \frac{\sum S_i^q}{T_q} \varphi_q \quad (1)$$

云资源在一定活跃范围内的基本状态主要取决于其自身的动态属性和最初基准的动态性。在获取云资源最初基准动态基础上,可获取云资源现存数据资源动态,即:

$$\eta_q^r = \mu_q^r S_i = \frac{\sum S_i}{T_q} \varphi_q \quad (2)$$

式中, η_q^r 代表云资源现存资源数据的动态; μ_q^r 代表现存云资源数据的特征。

云资源数据变化较为复杂,不能始终保持一成不变,故而在云资源中增加新的数据后,其云资源的动态特性必然发生改变,所以还需要对云资源的来源动态特性进行分析,对其动态特性进行修正^[9],进而提高云资源特性稳定性,即:

$$\psi S_i = \frac{\sum S_i (1 + \frac{\mu_q^r \varphi_q}{T_q})}{T_q + 1} \varphi_q \quad (3)$$

式中, ψS_i 代表云资源数据来源的动态基准。

在上述分析中,云资源动态特性的基准对资源的最初基准以及后期变换后的特性均进行有效保持,为云资源动态负载均衡控制奠定基础。

1.2 云资源动态负载特性采样

在上述云资源动态特性分析基础上,对云资源动态负载特性进行采样。云资源的使用中,使用者希望访问的云资源可覆盖的范围更大,这样可实现云资源利益的最大化。因此,根据云资源动态负载特征选择云资源,以实现云资源动态负载均衡控制。

基于云资源动态特性的采样方法主要通过云资源节点的负载和组户数决定^[10]。假设云资源节点负载范围为 $[0, N]$,在该范围内任意选取对节点负载有影响的使用者,则此时云资源节点负载组合被囊括在内,如图 1 所示。

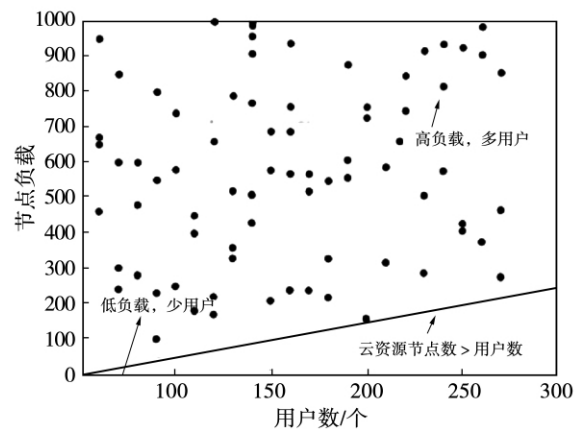


图 1 云资源节点负载采样

图1中,当云资源节点上租户的数量发生变化时,其节点的总负载也随之发生变化。云资源节点的负载是根据租户的变化决定,在云资源节点负载采样中可以覆盖较大范围。

实际上云资源上的每个节点均需要处理多种不同复杂程度租户的访问,此时,云资源节点负载发生波动,其均衡性被破坏。为了保持云资源的负载均衡,需要对云资源节点设置负载上限和下限。其上限是指能够满足所有租户的要求前提下,能够处理的最大限度;其下限是指租户请求未能达到的此值的未过载状态。因此,云资源节点负载范围应设置为 $[M_{\min}, M_{\max}]$,减少云资源节点空间的负载,以保证云资源节点的负载均衡性,修正后的云资源节点负载采样结果,如图2所示。

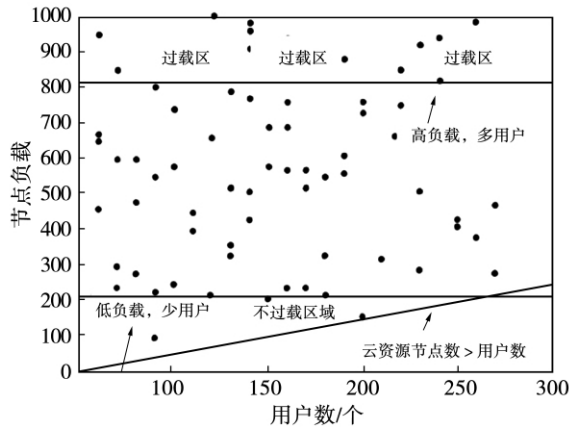


图2 修正后云资源节点负载采样

在云资源节点负载采样后,对其节点的负载均衡进行控制。假设云资源节点数量为 N 个,其任务数量为 H 个,量化云资源节点的动态负载均衡属性和任务属性^[11],在 j 时刻下,云资源节点的访问率为:

$$F(j) = \frac{\theta(\kappa, j)}{\vartheta(\kappa)} \quad (4)$$

式中, $\theta(\kappa, j)$ 代表云资源节点 κ 在 j 时刻处理访问的次数; $\vartheta(\kappa)$ 代表云资源节点 κ 可共同处理访问的总次数。

在上述分析基础上,可以获取云资源整体被同时访问的概率,该值为云资源节点负载阈值^[12],根据此阈值决定云资源动态负载均衡状态,即:

$$\tau = \frac{\sum_{j=1}^N F(j)}{N} \quad (5)$$

根据式(5)实现云资源动态负载均衡控制。为云资源的智能动态伸缩实现提供保障。

2 增强学习算法的云资源弹性收缩实现

在上述云资源动态负载均衡控制基础上,引入增强学习算法对云资源进行弹性收缩,增强云资源的高度弹性和优化使用。

基于增强学习算法的云资源弹性收缩中,将在云资源节点上增加不同的权重值,采用非均匀离散傅里叶变换方法,重新获取云资源节点样本,对其进行均匀调整,即:

$$\sigma_l(\lambda) = \int_v \lambda \rho(\chi) d\chi \quad (6)$$

式中, $\sigma_l(\lambda)$ 代表重新获取的云资源节点样本密集程度值; χ 代表云资源节点样本; v 代表云资源节点总数的权重值大小; c 代表云资源节点样本权重值的大小。

在获取全新云资源节点样本后,采用递归处理方式进行加权计算,获取云资源节点的状态:

$$Z_a = \frac{\rho(\chi_1) + \rho(\chi_2) + \dots + \rho(\chi_n)}{n} \quad (7)$$

式中, Z_a 表示云资源节点样本的递归值; $\rho(\chi_1)$ 、 $\rho(\chi_2) \dots \rho(\chi_n)$ 代表云资源节点对应的权重值; n 表示云资源节点总数。

获得云资源节点状态后,本文采用增强学习算法中的 Q 学习方法,实现云资源的弹性收缩^[13]。假设给定一个云资源伸缩策略 \bar{h} ,定义 Q 学习方法中的函数在云资源状态为 d 时,对其采取伸缩策略后的期望伸缩值为:

$$Q_{l+1} = \varpi + \delta \sum_{l+1 \in g} [R(g_l, d, g_{l+1})] \times Y(g_{l+1}, \bar{h}) \quad (8)$$

式中, δ 代表伸缩因子; g 代表云资源节点当前负载状态; $R(g_l, d, g_{l+1})$ 代表云资源在负载为 R 时,云资源动态伸缩后的状态概率; $Y(g_{l+1}, \bar{h})$ 代表云资源节点状态开始到执行伸缩策略 \bar{h} 后的伸缩期望值,该值为云资源根据负载状态进行自动伸缩的最优值。

云资源节点的负载随时间的变化不断发生变

化,难以始终保持在平衡状态,所以云资源节点负载当前状态需要被观察,不断自适应选择执行弹性收缩的策略,并及时将弹性收缩值进行回馈,根据式(9) 不断调整云资源弹性收缩值,即:

$$Q_{l+1}(g,e) = (1 - \beta) Q_{l+1}(g,e) + \beta \varpi + \delta \max_{e \in g} Q_{l+1}(g_{l+1},e) Q_{l+1}(g,e) \quad (9)$$

$$(g,e) \neq (g_l,e_l)$$

式中, β 代表学习效率; e 代表变动的云资源负载值。

在满足一定条件下,云资源的弹性收缩值在不同均衡负载下可以采用式(9) 进行多次迭代, $Q_{l+1}(g,e)$ 的概率活跃在 $[0,1]$ 之间,且接近 1 或 0 时,表示云资源节点负载均衡达到上限或下限,此时云资源进行自动的弹性伸缩。

3 实验分析

3.1 实验环境

为了验证所提方法的科学有效性,进行实验分析。实验选择在 CLouldStone 环境下进行,实验操作系统为 Windows XP 系统,配置 16 核 CPU,系统运行内存为 8GB。

3.2 实验参数

实验参数如表 1 所示。

表 1 实验参数

| 参数 | 取值 |
|-----------|-------------|
| 网络带宽/Mb/s | 1000 |
| 云资源节点/个 | 600 |
| 云资源访问间隔/s | 1 |
| 任务长度/mips | [1500,4000] |
| 迭代次数/次 | 50 |

3.3 实验结果分析

3.3.1 不同方法云资源利用效率分析

为了验证所提方法的科学有效性,实验对比了所提方法、文献 [4] 方法以及文献 [5] 方法对云资源进行弹性收缩后的资源利用效率,实验结果如图 3 所示。

分析图 3 可知,在相同实验环境下,通过三种方法弹性伸缩后的资源利用效率存在一定差异。其中,所提方法的资源利用效率较高,最高可达约

97%,文献 [4] 方法和文献 [5] 方法的资源利用效率最高分别约为 86% 和 57%,相比之下,所提方法的资源利用效率较高。这是由于所提方法在进行弹性压缩之前对云资源的动态特性进行详细分析,并对其节点的均衡负载能力进行控制,进而提高了所提方法的资源利用效率。

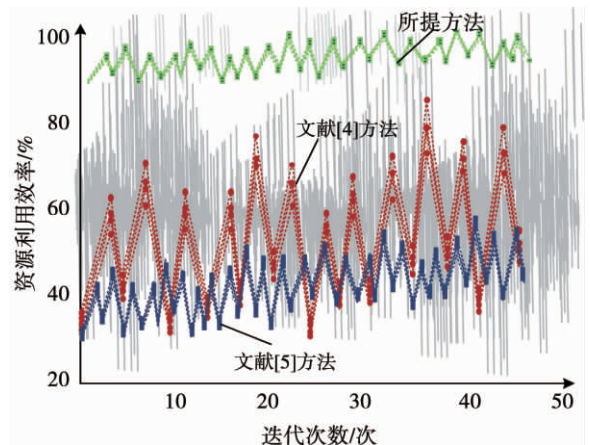


图 3 不同方法资源利用效率对比

3.3.2 不同方法云资源的负载均衡性分析

云资源的负载均衡是衡量云资源弹性收缩后其性能的重要指标,实验对比了所提方法、文献 [4] 方法以及文献 [5] 方法的负载均衡数值,其中,数值越大说明该方法的负载均衡性能更佳,实验结果如图 4 所示。

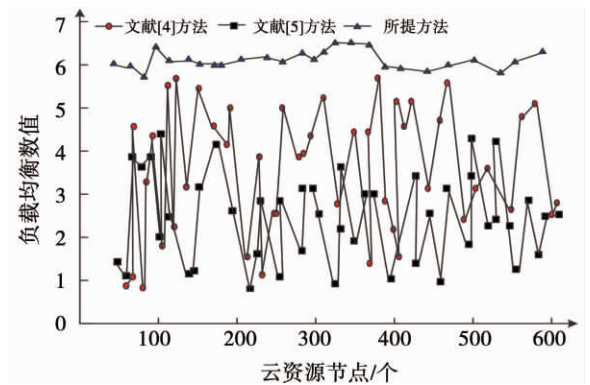


图 4 不同方法云资源的负载均衡性对比

从图 4 中可看出,所提方法、文献 [4] 方法以及文献 [5] 方法随着云资源节点数量的不断变化,三种方法的负载均衡性数值也随之发生变化。其中,所提方法的负载均衡数值始终高于其他两

种方法,且波动幅度相比之下较小,最大值约为6.5,而其他两种方法的波动幅度较大,且最大值分别约为5.7和4.6,验证了所提方法的负载均衡性能较好。

3.3.3 不同方法云资源的调度精度分析

为了进一步验证所提方法的可靠性,实验分析了三种方法进行弹性压缩后云资源的调度精度,实验结果如图5所示。

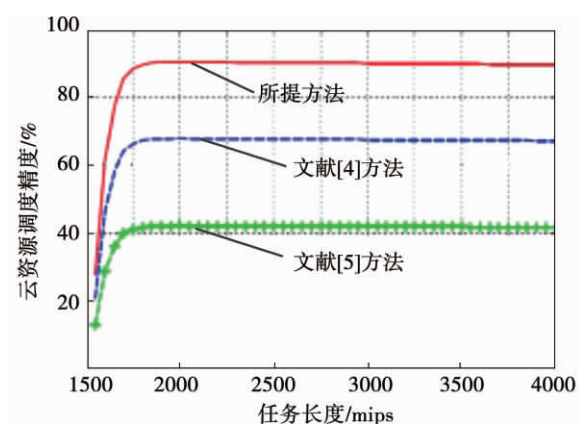


图5 不同方法云资源调度精度对比

分析图5可知,在任务长度相同情况下,三种方法云资源调度的精度存在一定差异。所提方法的调度精度最高约为90%,文献[4]方法的资源调度精度最高约为69%,文献[5]方法的资源调度精度最高约为42%,三种方法中所提方法的资源调度精度最好。这是由于所提方法在进行云资源弹性收缩时采用增强学习算法不断迭代最优策略,在迭代中提高了云资源的调度精度,验证所提方法的可靠性。

4 结束语

针对云资源使用量较大且负载能力有限等问题,提出基于增强学习算法的云资源动态负载均衡弹性伸缩方法。与传统方法相比,所提方法在弹性收缩后的资源利用率明显提升,且其负载均衡性能较好,具有一定可行性。但本文还存在诸多不足,未来将从更细致的环节中完善云资源弹性收缩的方法。

参考文献:

- [1] 胡声秋,吴玲丽.一种云计算环境下海量数据安全特征提取算法[J].信息技术,2019,43(1):93-96.
- [2] 王明青,杨博文,杨坚.LTE可伸缩视频组播的动态资源分配算法[J].计算机工程,2018,44(10):280-286.
- [3] 喻德旷,杨谊,钱俊.云计算资源的动态随机扰动的粒子群优化策略[J].计算机应用,2018,38(12):3490-3495.
- [4] 彭丽苹,吕晓丹,蒋朝惠,等.基于Docker的云资源弹性调度策略[J].计算机应用,2018,38(2):557-562.
- [5] 王峰.云计算下安全可控的移动弹性资源动态调度算法[J].科学技术与工程,2018,18(3):291-296.
- [6] 王春娟.基于蚁群算法的物联网链路负载均衡控制研究[J].信息技术,2020,44(6):121-124,129.
- [7] 李树全,刘磊,朱大勇,等.一种面向云存储的数据动态验证方案[J].计算机科学,2020,47(2):256-261.
- [8] 徐彬,王传礼,喻曹丰,等.超磁致伸缩驱动微定位平台的动态特性[J].科学技术与工程,2018,18(34):71-77.
- [9] 刘益青,高伟聪,王林先,等.长数据窗算法的动态特性及其在快速保护中的应用[J].电力自动化设备,2018,38(1):199-204.
- [10] 谢小军,于浩,陶磊,等.基于动态数据压缩的能量采集无线传感网络数据收集优化[J].计算机应用,2018,38(8):2353-2358.
- [11] 刘春玲,王静,张然.异构型网络通信中负载均衡选择仿真研究[J].计算机仿真,2018,35(1):269-274.
- [12] 胡珂,李成名,沈建明.分布式时空数据库去中心化负载均衡方法[J].测绘通报,2018(12):65-68,128.
- [13] 童钊,邓小妹,陈洪剑,等.云环境下基于强化学习的多目标任务调度算法[J].小型微型计算机系统,2020,41(2):285-290.

(责任编辑:杨静)