

DOI:10.11918/j.issn.0367-6234.201711104

随机需求下面向异质费用的云资源调度算法

刘 扬¹, 魏 蔚¹, 张伟哲²

(1. 河南工业大学 信息科学与工程学院, 郑州 450001; 2. 哈尔滨工业大学 计算机科学与技术学院, 哈尔滨 150001)

摘要: 采用分布式云构建流媒体服务等高资源消耗系统,既符合应用多区域部署的要求,也能充分利用云中资源保证服务质量,同时还能进行系统预算成本控制.由于各区域云中心费用函数存在差别,分布式云中调度需引入异质费用模型,结合流媒体应用中用户请求高度动态随机的特征,在给定的费用预算下响应尽可能多的用户请求.均值需求模型忽略了资源需求在短时间间隔内的变化细节,导致资源利用率低下.为克服均值需求模型的缺点,采用随机需求模型以捕捉细粒度资源需求,使用通用代价函数描述异质费用模型,建立更具通用性的非线性规划问题模型;为降低求解算法的复杂度,基于动态规划快速获得解的下界,再迭代逼近获取近优解.实验结果表明:相比经典的基于均值的调度算法,在区域数量较大时,平均能额外满足15%的用户请求;随着预算的减少,能额外满足近40%的用户请求;且不受各区域价格函数差异和用户访问需求差异的影响.因此,在构建全球部署的大规模流媒体服务系统时,算法能以较低的计算代价显著增加响应的用户请求量,广泛适应各种不同的云基础设施服务提供商.

关键词: 随机需求;资源调度;非线性规划;异质费用模型;云计算

中图分类号: TP301 **文献标志码:** A **文章编号:** 0367-6234(2018)11-0116-06

Heterogeneous cost oriented cloud resource scheduling algorithm for stochastic demand

LIU Yang¹, WEI Wei¹, ZHANG Weizhe²

(1. College of Information Science and Engineering, Henan University of Technology, Zhengzhou 450001, China;
2. School of Computer Science and Engineering, Harbin Institute of Technology, Harbin 150001, China)

Abstract: It is recommended to use a distributed cloud to construct a high resource consumption system such as streaming media service, which not only meets the requirements of multi-region deployment, but can also make full use of the resources in the cloud to ensure the service quality while controlling system budget costs. Make the difference of cost functions in cloud centers sitting in different regions, the heterogeneous cost model should be introduced in distributed cloud oriented scheduling. Given the highly dynamic and random characteristics of user requests in streaming media applications, it is expected to respond to as many user requests as possible under a given cost budget. Mean demand model ignores details of changes in resource requirements over short time intervals, and leads to inefficient use of resources. To overcome the disadvantages of mean demand model, we use a stochastic model to capture the fine-grained information of resource demand, and use a common cost function to describe the heterogeneous cost model, and also establish a general nonlinear programming problem model. To reduce the algorithm complexity, the lower bound of the solution is quickly obtained based on dynamic programming, and then the near optimal solution is obtained by iteratively approximation. The simulation results show that, compared with the classical average demand model based scheduling algorithm, the proposed algorithm can additionally meet 15% more of the user requests when the number of regions is large, and can additionally satisfy up to nearly 40% of the user requests as the budget decreases. The algorithm is not affected by differences in price functions or by user requests statistics across different regions. As a result, when used in globally deployed large-scale streaming media system, the proposed algorithm can significantly increase the number of satisfied user requests with limited computing time, thus is adaptable to a wide range of cloud infrastructure service providers.

Keywords: stochastic demand; resource scheduling; nonlinear programming; heterogeneous cost model; cloud computing

收稿日期: 2017-11-15

基金项目: 国家自然科学基金(U1504607, 61472460, 61702162); 河南省高校科技创新团队支持计划(17IRTSTHN011); 河南省教育厅科学技术研究重点项目(17A520004); 河南省科技厅科技攻关项目(172102110013)

作者简介: 刘 扬(1978—),女,副教授,硕士生导师;

张伟哲(1976—),男,教授,博士生导师

通信作者: 魏 蔚, weiwei_ise@haut.edu.cn

为满足海量资源需求并最小化资源使用费用,已有的大规模在线服务网站(如Netflix^[1]和YouTube^[2])通常基于云计算平台搭建系统(如使用Amazon EC2^[3]和Microsoft Azure^[4]).随着大规模应用系统的发展,为满足来自全球各地用户请求,需

将系统部署在散布在全球各区域的分布式云数据中心,就近响应用户请求以满足服务质量要求。云中资源分配方法通常将资源需求均值作为输入,采用传统的线性规划算法求解^[5-11]。但近几年的研究表明^[12-16]:在处理高度动态的随机资源需求时,随机需求模型由于保留更多的细粒度信息,比均值资源需求模型更加有效,相同资源量下可满足更多资源需求。但已有随机需求模型使得调度转化成非线性规划问题,相比均值模型算法已大大增加计算量,各区域由于基础设施上的差别,资源费用函数并不相同,若在优化目标中考虑各区域异质的非线性计费函数,会进一步增加调度算法复杂度。现有工作简化或省略费用模型,调度时仅考虑资源总量限制^[12-16],不符合云平台调度实际情况,不能发挥云中资源调度的优势。

本文尝试将通用的异质费用模型融入面向随机需求的调度中,建立普适性的问题形式,探索一种适用于跨区域分布式云平台的,不受各区域价格函数差异和用户访问需求差异影响的,低开销、高收益的资源调度方法。

1 通用资源分配问题的定义

为准确表示实际系统调度模型,引入两个时间尺度,分别是调度间隔 S^t 和需求测量间隔 S^d ,其中调度间隔 S^t 一般远大于需求测量间隔 S^d ,如常见云平台中虚拟机最小计费时间间隔一般为 1 小时^[3-4],使得对应的调度间隔也为 1 个小时,而随机需求测量间隔一般是 10 min。因此设 $S^t = IS^d$,即一个调度间隔分为 I 个需求测量间隔,并记录单个测量间隔内每隔如 5 s 的需求量,累计多个值获得测量间隔内需求的分布信息,基于历史数据预测未来某个测量间隔的需求分布作为调度算法的输入。设有来自 J 个区域的针对 I 种资源的用户请求,用 L_j 表示将在区域 j 中部署的资源数量。则在一个调度间隔中, J 个区域所有分配构成一个资源配置方案,记为 $P = \{L_j\}$ 。对来自区域 j 的请求,若可满足,则它将分配到:位于区域 j 的资源,称之为本地满足;或位于其它区域的资源,称之为远程满足。若请求未被分配到任何资源,则将其标记为“未满足”。注意请求是否在本地区域将会影响服务质量,一般认为本地满足时由于实际地理距离较短,请求延时较低且受网络状态影响较少,可保证较高的服务质量。

对于单次调度中第 i 个需求测量间隔中区域 j 和所有区域的需求实际采样标量值分别为 g_j^i 和 g^i 。则所有区域的本地满足需求量的和为

$\sum_{j=1}^J \min(L_j, g_j^i)$,所有区域本地和远程满足的所有需求量的和为 $\min(L, g^i)$,其中 $L = \sum_{j=1}^J L_j$ 。则所有区域满足的非本地需求量的和为 $\min(L, g^i) - \sum_{j=1}^J \min(L_j, g_j^i)$ 。因此在需求测量间隔 i 内资源收益为本地满足需求量和非本地满足需求量的加权和,即 $w_1 \min(L, g^i) + w_2 \sum_{j=1}^J (\min(L, g^i) - \sum_{j=1}^J \min(L_j, g_j^i))$,通过合并同类项为全部满足需求量和本地满足需求量的加权和:

$$R^i = w_{\text{sat}} \min(L, g^i) + w_{\text{loc}} \sum_{j=1}^J \min(L_j, g_j^i). \quad (1)$$

则单次调度间隔的收益可表示为各个需求测量间隔中收益的和:

$$R = w_{\text{sat}} \sum_{i=1}^I \min(L, g^i) + w_{\text{loc}} \sum_{i=1}^I \sum_{j=1}^J \min(L_j, g_j^i). \quad (2)$$

云资源调度限制一般用预算表示,设为金额总量 C ,此时需考虑各个云数据中心内资源的价格,由于常见云基础设施提供商在不同地区的数据中心采用不同的定价策略,为使得算法具通用性,计费函数可为任意的非线性单调递减函数。设区域 j 的资源计费函数为 $f_j(\cdot)$,设单次调度前预测得知各测量时间段的需求量为 g_j^i 和 g^i ,在给定预算 C 下资源调度问题可为如下形式:

$$\max(R) \quad \text{s.t.} \quad \sum_{j=1}^J f_j(L_j) = C. \quad (3)$$

而实际场景中的预测算法无法获取精确的 g_j^i 和 g^i ,经典方法通过预测两者的均值参与求解,但丢失较多需求细节信息。现有预测算法可得到 g_j^i 和 g^i 的分布函数,因此在需求测量间隔 i 内,将区域 j 中所有用户对本地资源的随机需求为随机变量 G_j^i ,所有区域用户对本地和远程资源的随机需求为随机变量 G^i ,累积分布函数为 $\text{cdf}_{i,j}$ 和 cdf_i 。则求解目标为期望收益最大化:

$$\max(E(R)) \quad \text{s.t.} \quad \sum_{j=1}^J f_j(L_j) = C. \quad (4)$$

2 算法

2.1 面向随机需求和异质费用的云调度算法

费用总量限制问题在式 4 中,求解目标包含 $\min(\cdot)$ 这一类非线性函数,而约束中同样包含非线性函数,是一个高度复杂的非线性规划问题。在常见的商用云计算系统中上述问题输入数据量很大,一般会有资源类型和区域数的乘积 $I \times J \geq 10^4$ 。为寻找高效求解算法,需先简化求解目标。对于任意连续

的非负随机变量 x 与常量 M 有

$$E(\min(M, x)) = \int_0^M (1 - \text{cdf}_x(c)) dc. \quad (5)$$

式中 $\text{cdf}_x(\cdot)$ 为 x 的累积概率分布函数, 则式 4 中 $E(R)$ 可转化为如下的闭合形式:

$$E(R) = w_{\text{sat}} \sum_{i=1}^I \int_0^{L_i} (1 - \text{cdf}_i(n)) dn + w_{\text{loc}} \sum_{i=1}^I \sum_{j=1}^J \int_0^{L_{ij}} (1 - \text{cdf}_{i,j}(n)) dn. \quad (6)$$

通过考察求解目标的结构, 发现该问题不存在给定费用限制下的全局最优解条件, 获取单个局部最优解的算法复杂度也很高, 因此求解算法采取以下步骤:

算法 1 面向随机需求和异质费用的云调度算法

- 1) 获取算法的预设参数.
- 2) 迭代调整逼近资源总量限制下资源分配方案.
- 3) 在不同区域之间交换已分配资源, 并保证整体费用不变并增加收益.
- 4) 若无法找到可增大收益的交换区域, 输出分配方案作为最终解.

2.2 基于资源总量限制的预分配算法

如何获取符合资源总量限制的解成为影响算法效率的关键, 引入一个针对随机需求的类似问题, 该问题中资源限制由资源总量表示, 问题形式为

$$\max(E(R)) \quad \text{s.t.} \quad \sum_{j=1}^J L_j = U. \quad (7)$$

式中 U 即为资源总量, 例如私有中心服务器总量. 定义式 7 对应的问题为资源总量限制问题. 费用总量限制问题和资源总量限制问题的最优解存在一定关系, 可获取资源总量限制问题的最优解作为费用总量限制问题的初始解.

2.2.1 符合资源总量限制求解算法原理

费用总量限制问题和资源总量限制问题的关系如下所述:

推论 1 给定资源限制 U , 若对资源总量限制问题的最优解 $L'_j (1 \leq j \leq J)$ 满足 $\sum_{j=1}^J f_j(L'_j) = C$, 则费用总量限制问题的最优解 $L_j (1 \leq j \leq J)$ 满足 $\sum_{j=1}^J L_j \geq U$.

证明: 首先, 若对费用总量限制问题的最优解 $L_j (1 \leq j \leq J)$ 有 $\sum_{j=1}^J L_j = U'$, 则费用总量限制问题最优解的目标函数值必然不高于资源总量限制问题在限制 $U = U'$ 时最优解的目标函数值. 因为若假设

此时费用总量限制问题最优解的目标函数值大于资源限制 U' 下资源总量限制问题的目标函数值, 而此时费用总量限制问题最优解满足 $\sum_{j=1}^J L_j = U'$, 因此必然也是资源总量限制问题的可行解, 且可行解的目标函数值小于等于资源总量限制问题最优解的目标函数值, 这和假设相矛盾, 因此上述命题成立.

其次, 若 $U' < U$, 则资源总量限制问题在限制 U' 下最优解的目标函数值必然小于在限制 U 下最优解的目标函数值. 因为若资源总量限制问题在资源限制 U' 下获取最优解, 此时将 $U - U'$ 资源量分配到未满足需求的区域, 可得到资源总量限制问题在资源限制 U 下的初始解, 且目标函数值大于资源限制 U' 下的目标函数值, 因此上述命题成立.

在上述证明的基础上, 设给定费用预算 C 下费用总量限制问题的最优解 $L_j (1 \leq j \leq J)$ 有 $\sum_{j=1}^J L_j = U' (U' < U)$, 因为费用总量限制问题最优解的目标函数值必不会高于资源限制 U' 下资源总量限制问题最优解的目标函数值, 必小于给定资源限制 U 下资源总量限制问题最优解的目标函数值, 这和资源总量限制问题获得资源限制 U 下最优解相矛盾, 因此推论 1 成立. 据推论 1, 通过构造合适的资源总量限制问题, 可快速得到满足下限的初始解, 在此基础上持续调整获取逼近最优解.

2.2.2 资源预分配算法

资源总量限制问题是一个非线性规划问题, 为快速求解资源总量限制问题, 引入如下定理:

定理 1 有 N 个函数. $F_1(x_1), F_2(x_2), \dots, F_N(x_N)$, 满足 $x_n \in [0, \infty)$ 和 $F_n(x_n) \geq 0 (1 \leq n \leq N)$. 其导函数是 $f'_1(x_1), f'_2(x_2), \dots, f'_N(x_N)$, 满足 $f'_n(x_n) \in [0, Y] (Y > 0)$ 且均是连续单调递减函数. 则对于约束 $\sum_{i=1}^N x_n = U, \sum_{i=1}^N f_i(x_i)$ 达到最大值时, 有 $f'_1(x_1) = f'_2(x_2) = \dots = f'_N(x_N)$.

证明: 定义 $f(X) = \sum_{i=1}^N f_i(x_i)$, 假定 $f(X)$ 取最大值时并不满足上述导数相等条件, 设 $f'_a(x_a) < f'_b(x_b) (a, b \in [1, N])$, 因 $f'_a(x_a)$ 和 $f'_b(x_b)$ 均为单调递减函数, 则必存在 x'_a 和 x'_b , 有 $x'_a > x_a$ 和 $x'_b < x_b$, 且 $x'_a + x'_b = x_a + x_b$ 并满足 $f'_a(x'_a) = f'_b(x'_b)$. 设 $f'_a(x'_a) = f'_b(x'_b) = Z$, 有

$$f_a(x'_a) - f_a(x_a) = \int_{x_a}^{x'_a} f'_a(n) > Z(x'_a - x_a), \quad (8)$$

$$f_b(x_b) - f_b(x'_b) = \int_{x'_b}^{x_b} f'_b(n) < Z(x_b - x'_b). \quad (9)$$

则有 $f_a(x'_a) - f_a(x_a) > f_b(x_b) - f_b(x'_b)$ 即 $f_a(x'_a) +$

$f_b(x'_b) > f_a(x_a) + f_b(x_b)$. 对任意不满足 $f'_1(x_1) = f'_2(x_2) = \dots = f'_N(x_N)$ 的 X , 必存在另外一个 X' 且 $f(X') > f(X)$, 即均不可能是最大值, 上述定理得证. 从而可快速获取资源总量限制问题的最优解.

推论 2 资源总量限制问题的最优解 $L'_j (1 \leq j \leq J)$ 满足 $\sum_{i=1}^1 \text{cdf}_{i,1}(L'_1) = \sum_{i=1}^1 \text{cdf}_{i,2}(L'_2) = \dots = \sum_{i=1}^1 \text{cdf}_{i,J}(L'_J)$

证明: 将资源总量限制问题可转化为 $F_j(L_j)$
 $= w_{\text{sat}} \sum_{i=1}^1 \int_0^L (1 - \text{cdf}_i(n)) dn / J + w_{\text{loc}} \sum_{i=1}^1 \int_0^{L_j} (1 - \text{cdf}_{i,j}(n)) dn$, 则导函数为 $f'_j(L_j) = w_{\text{loc}} (1 - \sum_{i=1}^1 \text{cdf}_{i,j}(L_j))$, 根据定理 1, 最优解必然满足推论 2 中的条件, 推论 2 得到证明.

根据上述推论, 若寻找给定预算 C 下费用总量限制问题的初始解, 只需快速遍历满足资源总量限制问题最优条件的解, 找到费用总和为 C 的解即可.

设对资源总量限制问题最优解有 $v = \sum_{i=1}^1 \text{cdf}_{i,1}(L'_1) = \sum_{i=1}^1 \text{cdf}_{i,2}(L'_2) = \dots = \sum_{i=1}^1 \text{cdf}_{i,J}(L'_J)$, 则对给定的 v , 最优解中 $L_j = h_j^{-1}(v)$, 其中 $h_j(L_j) = \sum_{i=1}^1 \text{cdf}_{i,j}(L_j)$. 构建 $h_j(\cdot)$, 再计算不同 v 值对应最优解的费用, 找到总费用为 C 的资源总量限制问题的最优解作为费用总量限制问题的初始解; 在不改变总费用 C 的基础上迭代调整各个 L_j 增加总的收益直到无法调整. 获取初始解的过程如下:

算法 2 基于资源总量限制的预分配算法.

输入: 需求累积分布函数 $\text{cdf}_i(\cdot)$, $\text{cdf}_{i,j}(\cdot)$ 及其反函数; 预算总量 C , 各区域资源的价格函数 $f_j(\cdot)$, 算法终止条件阈值 D .

输出: 各区域资源量初始解 L'_j .

- 1) $s = I/2, v = I$;
- 2) WHILE (TRUE) {
- 3) FOR ($j = 1$ to J) {
- 4) 求解 $\sum_{i=1}^1 \text{cdf}_{i,1}(L'_j) = v$ 以获得 L'_j ;
- 5) }
- 6) $d = \sum_{j=1}^J f_j(L'_j) - C$;
- 7) $s = s/2$;
- 8) IF ($|d| \leq D$) 输出初始解并返回;
- 9) IF ($d < 0$) $v = v - s$;
- 10) ELSE $v = v + s$;
- 11) }

算法 2 中, 通常情况下的值 $I = 6$, 则 2~11 行的

循环在运行 50 次后, $s \approx 10^{-15}$, 此时 9 和 10 行中 v 值在加上或减去 s 几乎不发生变化, 则下个循环中 3~5 行根据 v 值的求得的 L'_j 和前次循环中几乎不发生变化, 此时足够逼近最优解. 调度前已知 3~5 行中需要的累计分布函数, 事先表示为快速查询函数值的哈希表, 则 3~5 行的计算过程可转化为 $O(1)$ 的查询过程, 则整个函数在大约 50 次循环后即可获得对应的结果, 该算法的复杂度可优化到 $O(50)$ 左右.

2.3 费用总量限制下资源分配算法

根据算法 2 快速获取预分配结果 L'_j , 在此基础上快速调整以获取近优解, 主要思路是在不改变整体费用基础上, 在各个区域间交换资源以增大收益, 直到收益无明显增加为止, 如以下算法所示.

算法 3 费用总量限制下资源分配算法.

输入: 需求累积分布函数 $\text{cdf}_i(\cdot)$, $\text{cdf}_{i,j}(\cdot)$ 及其反函数; 预算总量 C , 各区域资源的价格函数 $f_j(\cdot)$, 迭代控制变量 M 和 N .

输出: 费用总量限制下的近优解 L'_j .

- 1) 基于算法 1 获取初始解 L'_j ;
- 2) FOR ($m = 1$ to M) {
- 3) $d = mC / (MJ)$;
- 4) $n = 0$;
- 5) FOR ($n = 1$ to N) {
- 6) 获取每区域 j 增加预算 d 后整体增加的收益, 保存为长度 J 列表 LA ;
- 7) 获取每区域 j 降低预算 d 后整体降低的收益, 保存为长度 J 列表 LD ;
- 8) IF ($\max(LA) > \max(LD)$) {
- 9) 在对应的区域间移动预算 d 并更新各区域资源量;
- 10) }
- 11) }
- 12) }

根据上述过程可知, 计算复杂度取决于循环参数 M 和 N , 循环次数越大则越逼近最优解, 具体取值需要在解的优化性和计算复杂度之间权衡.

3 实验

在各类场景中比较本文算法和均值算法的效果差异. 其中, 调度间隔 $S^1 = 1 \text{ h}$, 需求测量间隔 $S^d = 10 \text{ min}$, 即 $I = 6$. 和文献[12]中实验场景类似, 每区域用户请求产生的资源需求表示为随机需求模型且符合 zipf 分布, 即给定任意一个请求, 来自区域 j 的概率为 $j^{-e} / \sum_{n=1}^J n^{-e}$, 其中实数 $e (e > 0)$ 为 zipf 参数.

设所有区域总需求量均值为 λ ，则每区域需求符合均值为 $\lambda_j^{-e} / \sum_{n=1}^J n^{-e}$ 的泊松分布。

实验中各缺省参数如下设定：区域总数缺省值为 $J = 100$ ，价格模型中，缺省情况下各区域虚拟机单价相等；zipf 参数范围一般为 $0.4 \sim 1.6$ ，缺省值为折中数值 1.0 ；预算总额 C 缺省值为一个调度间隔中各区域恰好满足均值需求时资源费用总和。为测试不同迭代次数对效果的影响，对算法 3 中迭代控制变量 M 和 N 取三组不同的值，分别为 $(M = 1\ 000, N = 1\ 000)$ ， $(M = 500, N = 500)$ ， $(M = 100, N = 100)$ 。比较本文算法和基于均值算法的效果，所对比的均值算法的输入为各区域各需求测量间隔的均值 $E(G_j^i)$ ，目标是各区域供应资源量和需求均值比例相等，且满足预算总额 C 。

比较区域数量变化时两种方法之间的差别见图 1。横坐标是区域数量 J ，纵坐标是本文算法相比均值算法在资源收益上的增加比例。可以看到，随着区域数量的增加，本文算法和均值算法的收益差别有增大的趋势，由于算法通过在两个区域间互换配置寻找更优分配方案，更多的区域使得有更高的概率找到优化的交换方案，从而使得收益上升。这也表明算法通过增加收益，能部分补偿区域数量增加导致的计算量增加，且迭代次数 M 和 N 越大，算法效果越好。在 $(M = 500, N = 500)$ 时，在算法计算量和效果之间达到一个较好的平衡状态。

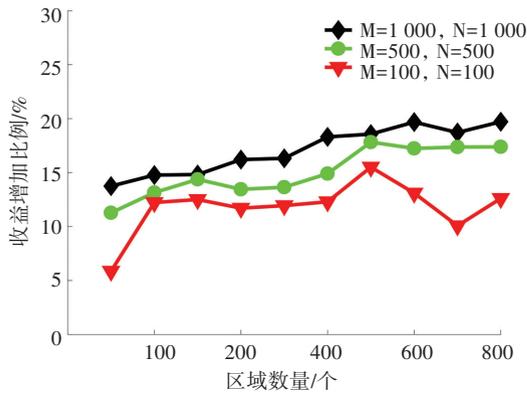


图 1 收益增加比例随区域数量的变化趋势

Fig.1 Percentage of increased revenue with different number of regions

由于实际环境中云基础设施提供商资源价格函数存在差别，甚至单个提供商管理的多个云数据中心之间价格也不相同，因此需研究各区域价格函数的差别对算法效果的影响。由于各提供商的各区域中价格差别一般不超过 2 倍^[1-4]，设每个区域均为线性函数，每个区域的最高和最低单价比率在 $1.0 \sim 2.0$ 之间变化，结果见图 2，横坐标是最高和最低单

价比率，纵坐标是本文算法相比均值算法在资源收益上的增加比例。可见价格差别扩大时，收益有波动并略有升高，但并没有明显的上升或下降趋势，意味着本文算法可适应常见各种类型的云平台，也可满足跨云提供商部署资源调度的需求。

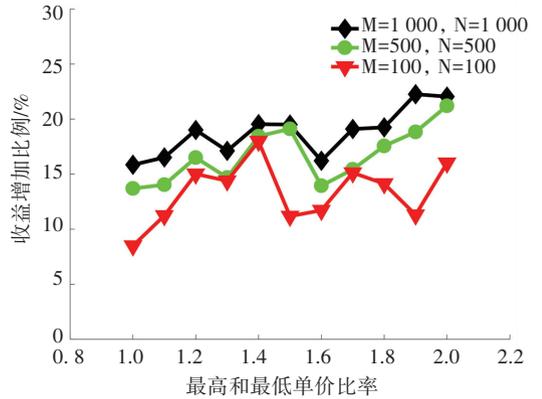


图 2 收益增加比例随价格比率区间的变化趋势

Fig.2 Percentage of increased revenue with different price range

由于常见的跨区域应用往往有一些热门的区域，即某些区域需求占较大比重，这种不同区域的热门程度的差别可由 zipf 参数刻画，zipf 参数值一般介于 $0.6 \sim 1.4$ 之间，参数值越大，表明各区域需求热门程度差别越大。这里比较不同参数值下收益的变化，结果见图 3，横坐标是 zipf 参数值，纵坐标是本文算法相比均值算法在资源收益上的增加比例。可见随着 zipf 参数的增加，资源收益略有降低，未发生较大变化。这意味着各区域需求热门程度的差别不会显著影响算法的效果，可适用于具不同跨区域需求差别的各类应用场景。大的迭代次数会提升算法效果，但 $(M = 500, N = 500)$ 和 $(M = 1\ 000, N = 1\ 000)$ 之间的差别不明显。

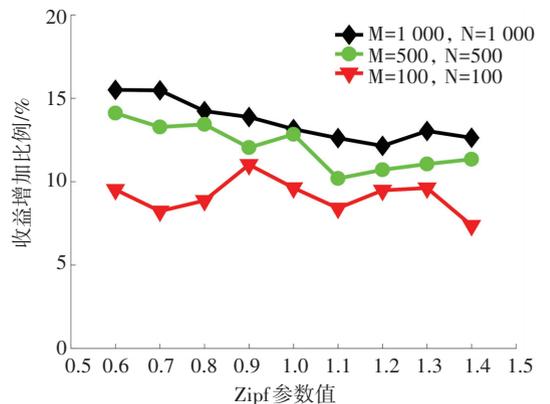


图 3 收益增加比例随 Zipf 参数的变化趋势

Fig.3 Percentage of increased revenue with different Zipf parameter
预算总额限制也会影响算法效果，因此比较不同预算额度下算法收益的变化，见图 4，横坐标的预

算以和均值预算总额的比例表示, 均值预算总额即每区域采用刚好满足需求均值的资源量对应的费用, 纵坐标是本文算法相比均值算法在资源收益上的增加比例. 可见随着预算的减少, 相对均值算法的额外收益迅速增加, 即在预算偏少的情况下, 比均值算法有更大的收益优势. 由于常见调度场景中预算相对不足, 因此本文算法可获得明显的优化效果.

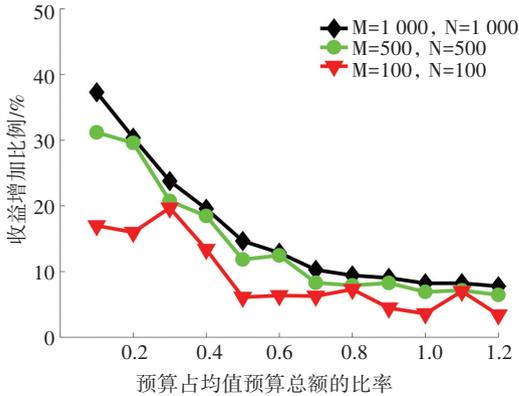


图 4 收益增加比例随预算额度的变化趋势

Fig.4 Percentage of increased revenue with different budget

4 结束语

本文研究分布式云平台中面向随机需求的通用资源分配问题, 算法考虑各地理区域间异质的费用模型, 采用预算总额的资源限制, 提出可获取近似解的快速求解方法. 相比已有的均值计算方法, 可额外满足近 40% 的资源需求, 因此可作为已有算法的有效补充.

参考文献

- [1] SUMMERS J, BRECHT T, EAGER D, et al. Characterizing the workload of a netflix streaming video serve [C]// Proc of IEEE International Symposium on Workload Characterization. Providence, USA: IEEE Press, 2016; 1. DOI: 10.1109/IISWC.2016.7581265
- [2] KHAN M L. Social media engagement: What motivates user participation and consumption on YouTube [J]. Computers in Human Behavior, 2017, 66(1): 236. DOI: 10.1016/j.chb.2016.09.024
- [3] EMERAS J, VARRETTE S, BOUVRY P. Amazon elastic compute cloud (EC2) vs. In-House HPC platform: a cost analysis [C]// Proc of International Conference on Cloud Computing. Hong Kong, China: IEEE Press, 2017; 1. DOI: 10.1109/TCC.2016.2628371
- [4] CARUTASU G, BOTEZATU M A, BOTEZATU C, et al. Cloud computing and windows azure [C]// Proc of International Conference on Electronics, Computers and Artificial Intelligence. Shenzhen, China: IEEE Press, 2017; 1. DOI: 10.1109/ECAI.2016.7861168
- [5] YOU Kun, TANG Bin, QIAN Zhuzhong, et al. Qos-aware placement of stream processing service [J]. The Journal of Supercomputing, 2013, 64(3): 919. DOI: 10.1007/s11227-010-0548-2
- [6] WANG Feng, LIU Jiangchuan, CHEN Minghua. Calms: Cloud-assisted live media streaming for globalized demands with time/region

- diversities [C]// Proc of IEEE Conference on Computer Communications. Orlando, USA: IEEE Press, 2012; 199. DOI: 10.1109/INFCOM.2012.6195578
- [7] NAN Xiaoming, HE Yifeng, GUAN Ling. Queuing model based resource optimization for multimedia cloud [J]. Journal of Visual Communication and Image Representation, 2014, 25(5): 928. DOI: 10.1016/j.jvcir.2014.02.008
- [8] ZHAO Yuhong, JIANG Hong, ZHOU Ke, and et al. Meeting service level agreement cost-effectively for video-on-demand applications in the cloud [C]// Proc of IEEE Conference on Computer Communications. Toronto, Canada: IEEE Press, 2014; 298. DOI: 10.1109/INFCOM.2014.6847951
- [9] 赵莉. 基于改进量子粒子群算法的云计算资源调度 [J]. 南京理工大学学报 (自然科学版), 2016, 40(2): 223. DOI: 10.14177/j.cnki.32-1397n.2016.40.02.015
- ZHAO Li. Cloud computing resource scheduling based on improved quantum particle swarm optimization algorithm [J]. Journal of Nanjing University of Science and Technology, 2016, 40(2): 223. DOI: 10.14177/j.cnki.32-1397n.2016.40.02.015
- [10] 夏庆兰, 兰雨晴, 唐甜等. 基于负载特征聚类的节能资源调度算法 [J]. 北京航空航天大学学报, 2015, 41(4): 680. DOI: 10.13700/j.bh.1001-5965.2014.0407
- XIA Qinglan, LAN Yuqing, TANG Tian, et al. Energy-saving resource scheduling algorithm based on workload characteristic clustering [J]. Journal of Beijing University of Aeronautics and Astronautics, 2015, 41(4): 680. DOI: 10.13700/j.bh.1001-5965.2014.0407
- [11] 张吉法, 胡斌, 徐东亮, 张小玉, 李卓球. 考虑丝束变形和铺层力学方向的铺放线型规划 [J]. 哈尔滨工业大学学报, 2016, 48(1): 172. DOI: 10.11918/j.issn.0367-6234.2016.01.026
- ZHANG Jifa, HU Bin, XU Dongliang, ZHANG Xiaoyu, LI Zhuoqi. Pattern planning for the deformation of fiber tows and mechanics direction of placement layers [J]. Journal of Harbin Institute of Technology, 2016, 48(1): 172. DOI: 10.11918/j.issn.0367-6234.2016.01.026
- [12] ROCHMAN Y, LEVY H, BROSH E. Resource placement and assignment in distributed network topologies [C]// Proc of IEEE Conference on Computer Communications. Turin, Italy: IEEE Press, 2013; 1914. DOI: 10.1109/INFCOM.2013.6566991
- [13] ROCHMAN Y, LEVY H, BROSH E. Efficient resource placement in cloud computing and network applications [J]. ACM SIGMETRICS Performance Evaluation Review, 2014, 42(2): 49. DOI: 10.1145/2667522.2667538
- [14] NIU Di, Xu Hong, Li Baochun, and et al. Quality-assured cloud bandwidth auto-scaling for video-on-demand applications [C]// Proc of IEEE Conference on Computer Communications. Orlando, USA: IEEE Press, 2012; 460. DOI: 10.1109/INFCOM.2012.6195785
- [15] WEI Wei, LIU Yang, ZHANG Yuhong. TRIMS: two-stage resource scheduling algorithm for cloud based live media streaming system [J]. IEICE TRANSACTIONS on Information and Systems, 2014, 97(7): 1731. DOI: 10.1587/transinf.E97.D.1731
- [16] WEI Wei, ZHANG Yuhong, LIU Yang, and et al. FRP: a fast resource placement algorithm in distributed cloud computing platform [J]. Concurrency and Computation Practice and Experience 2015, 28(5): 1399. DOI: 10.1002/cpe.3654