

云平台海量任务的多约束调度算法优化研究

曾 薇

(四川理工学院 计算机学院, 四川 自贡 643000)

摘 要: 提出一种新的云平台海量任务的多约束调度算法, 将任务优先级、资源服务能力、完成时间和费用作为约束条件建立多约束调度模型, 通过效用函数将云平台海量任务多约束调度模型转换成效用, 将最大化全部用户的总效用作为目标函数. 依据蚁群原理, 将云平台中被频繁调度的计算资源与数据资源之间的传输链路看作是蚂蚁经过的路径, 将信息素引入云平台任务调度中, 获取符合约束条件的计算资源, 达到调度目标最优. 实验结果表明, 此算法不仅效率高, 而且负载均衡性强, 所需总费用较低.

关键词: 云平台; 海量任务; 多约束; 调度

中图分类号: TP393

文献标识码: A

文章编号: 1000—7180(2016)06—0130—04

The Optimization of Many Constraints Scheduling Algorithm Based on Cloud Mass Task

ZENG Wei

(School of Computer Science, Sichuan University of Science and Engineering, Zigong 643000, China)

Abstract: Put forward a new cloud platform massive task constraint of scheduling algorithm, will be completed task priority, resource service capacity, time and cost as constraint condition to establish more constrained scheduling model, through the utility function will cloud platform with massive task constraint scheduling model transformation efforts, and to all users to maximize the total utility as the target function. According to the principle of ant colony, is frequently scheduling in the cloud computing and data resources transmission link between the route as ants, introducing a pheromone task scheduling in the cloud platform, access to conform to the constraints of computing resources, to achieve the optimal scheduling objectives. The experimental results show that the proposed algorithm has not only high efficiency, and the load balance is strong, the overall cost is low.

Key words: cloud platform; massive task; many constraints; scheduling

1 引言

随着计算机网络的逐渐发展, 云平台成为当前国内外计算机技术研究的关键, 被广泛应用于各个领域^[1,2]. 云平台海量任务调度为云计算技术的关键, 其主要用于对用户提交的任务进行最优调度, 使云平台的运行效率大大提高^[3,4]. 因此, 研究云平台海量任务调度方法具有重要意义, 它已经成为相关学者研究的重点课题^[5,6].

目前, 有关云平台海量任务调度的研究有很多, 主

要包括迪杰斯科拉法、模拟退火法、蚁群法等, 相关研究也取得了一定的成果. 其中, 文献[7]针对云平台节点稳定性差的缺点, 在保证云平台负载平衡和 CPU 空闲计算能力的前提下, 提出一种基于两级联合调度算法的云平台海量任务调度方法, 该方法随着数量的逐渐增加, 系统动态平衡效果也逐渐提高. 但该方法实现过程复杂, 不适用于实际应用. 文献[8]将任务完成时间看作是云平台任务调度的条件, 获取合理的资源执行任务, 得到目标函数, 采用神经网络法对目标函数进行求解. 该方法可减少任务完成时间, 但任务并行性与

资源利用率较低.文献[9]提出一种基于 Min-min 算法的云平台海量任务调度方法,该方法优先选择完成时间最短的任务进行调度,以最高的效率对调度任务进行调度,从而减少任务完成时间.然而该方法只考虑任务完成时间最快的局部最优解,会造成系统负载不均衡,特别是在任务完成时间之间的差异较大时,负面效应格外突出.文献[10]提出一种基于贪婪策略和最少优先策略的云平台海量任务调度方法,通过贪婪策略将节点缓冲区分割成两部分,一部分通过最少优先策略对云平台海量任务进行调度,其余部分则采用贪婪策略,然而该方法没有考虑新的云平台海量任务的调度,实时性与适应能力较低.

针对上述方法的弊端,本文提出一种新的云平台海量任务的多约束调度算法.

2 云平台海量任务的多约束调度算法优化研究

2.1 云平台海量任务多约束调度模型

云平台海量任务调度将任务优先级、资源服务能力、完成时间和费用作为约束条件建立多约束调度模型.为了达到用户的要求,在多约束模型中通过效用函数将约束变为效用,以描述用户的满意度.调度模型需最大化用户的效用而不是仅仅达到某个用户的最大效应,所以需从整个系统角度分析,使全部用户的总效用最大化.

2.1.1 任务优先级计算

任务优先级是云平台中任务重要程度的体现,本节综合分析任务的用户权限种类、紧急程度及长度对任务优先级进行计算.在此之前,首先对相关数据进行标准化处理.用 Y_{ij} 描述云平台海量任务中的任意任务 t_i 的第 j 个属性的标准化处理结果,则有

$$Y_{ij} = (x_{ij} - m_j) / S_j \quad (1)$$

式中, x_{ij} 用于描述任务 t_i 的第 j 个属性值; m_j 用于描述全部任务第 j 个属性的平均值; S_j 用于描述全部任务第 j 个属性的绝对偏移,公式描述如下:

$$S_j = \frac{1}{n} (|x_{1j} - m_j| + |x_{2j} - m_j| + \dots + |x_{nj} - m_j|) \quad (2)$$

用 $P(t_i)$ 描述任务 t_i 的优先级;用 t_{nr_i} 、 t_{m_i} 及 t_{nl_i} 分别描述对任务 t_i 属性 t_{right} 、 $t_{urgency}$ 与 t_{length} 的标准化处理结果,则通过加权法对任务优先级进行计算的公式如下:

$$P(t_i) = \epsilon_1 \times t_{nr_i} + \epsilon_2 \times t_{m_i} + \epsilon_3 \times t_{nl_i} \quad (3)$$

式中, $i = [1, n]$; $\epsilon_1, \epsilon_2, \epsilon_3 \in [0, 1]$ 为权重因子,同

时有 $\epsilon_1 + \epsilon_2 + \epsilon_3 = 1$.

2.1.2 资源服务能力计算

通过优先级的计算结果对任务进行排序,将高优先级任务调度到更优的资源上,使资源分配得到保证.

为了描述资源的能力,通过下式求出资源的总服务能力:

$$S(r_j) = \sqrt{(r_{nc_j})^2 + (r_{nb_j})^2 + (r_{ns_j})^2} \quad (4)$$

式中, $S(r_j)$ 用于描述资源 r_j 的总服务能力; r_{nc_j} 、 r_{nb_j} 与 r_{ns_j} 分别用于描述资源 r_j 的计算能力、通信能力及存储能力的数据标准化结果.

2.1.3 完成时间和费用计算

最大程度地减少云平台下海量任务的完成时间与执行费用,为增强云平台服务质量的关键.云平台中任务的完成时间、执行费用的计算方法如下.

时间-费用折中是对任务 t_i 在资源 r_j 上的执行时间与费用的综合处理结果.用 ρ_{ij} 描述时间-费用折中值,则有

$$\rho_{ij} = \omega_1 \times \frac{ct_{ij}}{avpc_i} + \omega_2 \times \frac{pt_{ij}}{avpc_i} \quad (5)$$

式中, ω_1, ω_2 用于描述任意任务在云平台资源上的完成时间及价格的加权因子,同时有 $\omega_1 + \omega_2 = 1$; $avpc_i$ 用于描述任务 t_i 的预期平均费用, $avpc_i = \frac{1}{m} \sum_{j=1}^m pc_{ij}$, m 为资源总数, pc_{ij} 为任务 t_i 在资源 r_j 上的理想费用; ct_{ij} 用于描述任务 t_i 在资源 r_j 上的理想完成时间, $ct_{ij} = pt_{ij} + st(j)$, $st(j)$ 用于描述资源 r_j 的最早可用时间; pt_{ij} 用于描述任务 t_i 在资源 r_j 上的理想完成时间, $pt_{ij} = t_{length_i} / r_{comp_j}$, t_{length_i} 用于描述任务 t_i 的长度; r_{comp_j} 用于描述资源 r_j 的计算能力.

随着 ρ_{ij} 值的逐渐增加,该任务在资源中的完成时间及价格逐渐升高;反之降低.

2.1.4 多约束模型的建立

依据上述分析,建立的云平台海量任务多约束调度模型如下:

$$E = \frac{P(t_i)S(r_j)}{\rho_{ij}} \quad (6)$$

由于算法分析了多约束条件而且是在云平台下,因此,需考虑用户的服务需求.通过效用函数 $U_i(q_i)$ 将云平台海量任务多约束调度模型转换成效用,则任务 t_i 的效用函数可描述成:

$$U_i(q_i) = \sum_{j=1}^{d_i} \omega_i^j U_i^j(q_i^j) \times \frac{P(t_i)S(r_j)}{\rho_{ij}} \quad (7)$$

式中, $U_i^j(q_i^j)$ 用于描述和任务 t_i 相应的第 j 个属性

q_i^j 的效用值; w_i^j 用于描述和任务 t_i 第 j 个属性相应的权重. 本节云平台海量任务调度的目标就是使全部用户的总效用 $\sum U_i^j(q_i^j)$ 达到最大.

2.2 基于蚁群算法的求解过程

通过蚁群算法对多约束调度模型进行求解的基本思想如下: 依据蚁群原理, 将云平台中被频繁调度的计算资源与数据资源之间的传输链路看作是蚂蚁经过的路径, 将信息素引入云平台任务调度中. 在进行初始化的过程中, 将蚂蚁分布于不同资源上, 即在不同资源中添加信息素, 根据资源的调度状态对各资源信息素进行更新, 在选择下一个任务的过程中, 除了优先选择信息素浓度较高的路径外, 还需考虑数据发送效率高的任务, 以获取符合约束条件的计算资源, 达到调度目标最优.

第 k 只蚂蚁选择下一个任务 t_j 的概率可描述成:

$$p_k^j(t) = \begin{cases} \tau_{ij}(t) \alpha \cap \theta_{ij} \beta / \sum_{s \in \text{allowed}_k} \tau_{is}(t) \alpha \cap \eta_{is} \beta, & j \in \text{allowed}_k \\ 0, & \text{else} \end{cases} \quad (8)$$

式中, α, β 为两个参数, 分别描述了蚂蚁在移动时所积累的信息域启发信息在选择路径中的相对重要性, 也就是信息素轨迹的重要性能和能见度的重要性; τ_{ij} 用于描述任务 t_i 和 t_j 之间路径的信息素浓度, 公式描述如下:

$$\tau_{ij} = (1 - \rho) T_{\text{comp}}(t_k, r) + \rho \Delta \tau_{ij} \quad (9)$$

式中, ρ 用于描述信息素挥发系数; $\rho \Delta \tau_{ij}$ 用于描述蚂蚁所留的信息素增量;

$$\Delta \tau_{ij} = \mu(p_{rc} + p_{rd}) / (b_{js} d_{js} l_{js}) \quad (10)$$

式中, μ 为常数; p_j 用于描述云平台任意任务 t_j 被执行完成的概率; p_{rd} 用于描述云平台任意任务 t_j 与资源 r_d 之间完成数据传输的概率; b_{js}, d_{js}, l_{js} 分别描述云平台任意任务 t_j 与资源之间的瓶颈带宽、传输时延和丢包率.

在进行循环迭代时, 仅采用局部最优解对信息素进行更新即可, 从而使蚂蚁更有针对性, 不仅避免了蚂蚁收敛至同一路径, 而且也提高了蚂蚁的搜索的速度, 在既定迭代次数后通过全局最优解对信息素进行更新. 为了防止陷入局部最优, 为路径的信息素浓度设定一个上下阈值, 从而防止某条路径上的信息素浓度远远超过其余路径.

采用蚁群算法对多约束模型总效用最优解进行求解的详细过程如下所示.

(1) 首先为所有任务路径 (r, s) 设置信息素强度初值, 用 R 描述最大搜索周期数, 用 I 描述既定迭

代次数, 用 m 描述蚁群数量. 将其随机引入 n 个任务中, 一次搜索的最多时间约束用 T 进行描述, 最优化目标函数值 $U_i(q_i)$ 是无穷大, 解集是空;

(2) 假设本次搜索时间函数 t 为 0, 开始计时. 依据云平台任务优先级、资源服务能力、时完成间与费用的当前值对任务进行调度, m 只蚂蚁从初始任务开始, 相互独立地随机运动, 通过式 (8) 的概率公式选择任务或本次搜索时间 $t = T$;

(3) 求出一个周期内全部蚂蚁走过的路径目标函数值:

$$y(p_k) = \sum U_i(q_i^j)(p_k^j) \quad (11)$$

式中, p_k^j 用于描述本次循环中第 k 只蚂蚁从初始节点到目的节点所走的路径. 如果第 k 只蚂蚁在本次搜索中没有到达目的节点, 则令其为无穷大;

(4) 对 $\max\{y(p_k^j)\}$ 进行计算, 若 $\max\{y(p_k^j)\} > Y$, 则令 $Y = \max\{y(p_k^j)\}$. 如果当前的迭代次数为既定迭代次数 I 的整数倍, 则通过全局最优路径对解集进行更新; 反之, 选择本次迭代中解的最大值, 通过该解值对解集进行更新;

(5) 令 $r = r + 1$, 当 $r < R$ 时, 重新进行步骤 (2), 继续进行迭代; 反之, 输出当前最优解.

3 实验结果分析及结论

为了验证本文提出的云平台海量任务的多约束调度算法的有效性, 需要进行相关的实验分析. 云平台任务调度实时框架用图 1 进行描述, 实验采用 Cloudsim 仿真平台对图 1 描述的过程进行仿真, 将最小最大算法作为对比进行分析.

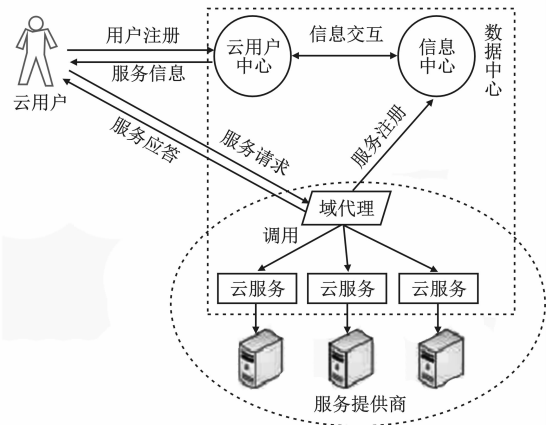


图1 云平台任务调度过程

随着云平台待调度任务数量的逐渐增加, 采用本文算法和最小最大算法所需的完成时间均逐渐增加, 但本文算法在完成时间上较最小最大算法有明

显的下降,且随着任务数量增长,下降趋势越来越明显.

本文算法的负载均衡性较高,这是因为本文算法不仅将完成时间作为约束条件,而且在任务调度中还进行了任务资源服务能力约束,在达到资源优化分配的情况下,也使云平台任务更均衡地调度到不同的资源上,增强了云平台的负载均衡性.

采用本文算法的服务总费明显低于最小最大算法,这是由于在对云平台海量任务进行调度时,本文算法将费用作为任务调度的约束条件,使其最终总费用较低.

4 结束语

本文提出一种新的云平台海量任务的多约束调度算法,将任务优先级、资源服务能力、完成时间和费用作为约束条件建立多约束调度模型,通过效用函数将云平台海量任务多约束调度模型转换成效用,将最大化全部用户的总效用作为目标函数.依据蚁群原理,将云平台中被频繁调度的计算资源与数据资源之间的传输链路看作是蚂蚁经过的路径,将信息素引入云平台任务调度中,获取符合约束条件的计算资源,达到调度目标最优.实验结果表明,所提算法不仅效率高,而且负载均衡性强,所需总费用较低.

参考文献:

[1] 朱映映,陈阳,明仲.云系统中面向海量多媒体数据的动态任务调度算法[J].小型微型计算机系统,2013,34(4):783-788.

[2] 姚卫红,黄小远,方仁孝.基于车联网应用的云平台任务调度算法[J].计算机仿真,2015,31(10):165-169.

[3] 吴洲.云计算环境下基于用户和资源约束的免疫效用均衡任务调度算法[J].计算机系统应用,2015,24(10):176-180.

[4] 刘丹琦.移动云计算环境下多 DAG 工作流的节能调度算法及能耗测量研究[D].乌鲁木齐:新疆大学,2014.

[5] 熊磊,陈宏伟,王淑平.CloudSim 中基于偏序关系的调度算法研究[J].湖北工业大学学报,2013,28(5):79-81.

[6] 徐惠萍,刘光明,孙涛,等.基于云计算技术的电信经营分析系统中对海量数据的研究[J].计算机光盘软件与应用,2013,18(12):267-269.

[7] 王祺元,闫宏印.基于改进 RM 算法的云环境任务调度研究[J].计算机测量与控制,2013,21(6):1612-1614.

[8] 马艳,龚斌,邹立达.网格环境下基于复制的能耗有效依赖任务调度研究[J].计算机研究与发展,2013,50(2):420-429.

[9] 何鹏举,吴来斌,宋凯华,等.基于粒子群算法的嵌入式云计算资源调度[J].电子设计工程,2014,22(10):88-90.

[10] 王鹏,黄焱,李坤,等.云计算集群相空间负载均衡度优先调度算法研究[J].计算机研究与发展,2014,51(5):1095-1107.

作者简介:

曾薇 (1981-),女,硕士研究生,讲师.研究方向为算法设计及优化、数据挖掘. E-mail:389713653@qq.com.

(上接第 129 页)

[5] 方薇,何留进,孙凯,等.采用元胞自动机的网络舆情传播模型研究[J].计算机应用,2010,30(3):751-755.

[6] 聂哲,李粤平,温晓军,等.个体间相互影响的网络舆情演变模型[J].计算机工程应用,2009,45(14):220-222.

[7] 陆题佳.因特网中危机信息传播规律及应对模式研究[D].合肥:中国科学技术大学,2010.

[8] Zeng Jianping, Zhang Shiyong, Wu Chengrong, et al. Modelling topic propagation over the Internet [J]. Mathematical and Computer Modelling of Dynamic Systems,2009,15(1):83-93.

[9] 王杨,尤科本,王梦瑶,等.基于博弈论的网络社区舆情

传播模型[J].计算机应用研究,2013,30(8):2480-2482.

[10] Hofbauer J, Sigmund K, Evolutionary games and population dynamics[D]. cambridge: Cambridge University press,1998.

[11] Hauert C, Szab G, Game theory and Physics[J]. Am J Phys,2005,73(5):405-414.

[12] Nowak M A, May R, Evolutionary games and spatial-chaos[J]. Letters to Natwre,1992,359(11):826-829.

作者简介:

黎永壹 男,(1979-),硕士,高级工程师.研究方向为信息系统与知识发现. E-mail:qzxylyy@163.com.