

doi:10.3772/j.issn.1002-0470.2018.11-12.003

基于新型休眠机制的云任务调度策略的研究^①

王秀双^② * * * * * 金顺福^③ * * * * *

(* 燕山大学信息科学与工程学院 秦皇岛 066004)

(** 河北省计算机虚拟技术与系统集成重点实验室 秦皇岛 066004)

(*** 河北省软件工程重点实验室 秦皇岛 066004)

摘 要 为提高云系统节能水平,同时满足云用户响应性能,研究了云任务调度策略及性能优化问题。引入休眠延迟机制与休眠唤醒阈值,提出一种基于新型休眠机制的云任务调度策略。根据云任务调度策略及云用户的随机行为,考虑物理机的唤醒过程,构建带有休假延迟与启动过程的同步 *N*-策略多重休假排队模型。利用拟生灭过程与矩阵几何解等数学工具,从云任务的平均逗留时间与云系统的节能水平等方面评估云任务调度策略的系统性能。进行系统实验,验证策略的有效性及模型解析的合理性,并揭示云用户响应性能与云系统节能水平之间的折衷关系。通过动态改变交叉概率与变异概率,改进遗传算法,给出云任务调度策略的智能优化方案,实现系统成本的最小化。

关键词 云计算, 任务调度, 休眠延迟, 唤醒阈值, 排队模型, 启动过程, 智能优化

0 引 言

云计算独有的计算能力引起业界的普遍关注。云计算用户需求的增加,使得云数据中心的规模越来越大,能源消耗问题也日渐突出^[1,2]。为了节约云数据中心能耗,降低云服务提供商的运营成本,同时保证云用户的服务质量,如何设计行之有效的云任务调度策略成为云计算领域的研究热点。

Vakilinia 等^[3]以最小化云数据中心总功耗为出发点,建立了虚拟机放置/迁移的平台。在优化调度及估计技术的协同作用下,最大限度减少直流输电的功耗。Yang 等^[4]从云计算能量管理出发,基于博弈理论,考虑平衡任务可靠性,提出一种任务调度算法,并建立计算节点的任务调度模型。在上述节能方案中,忽略了空闲能耗对系统节能水平的影响。

考虑云资源的低利用率,让空闲虚拟机进行深

度休眠可有效减少能量消耗。Jin 等^[5]针对云数据中心耗能问题,提出具有速率切换与多重休眠模式的节能策略。Guo 等^[6]提出一种动态优化云计算中能量与性能的启发式算法。使用虚拟机的实时迁移技术,通过动态合并虚拟机,将空闲服务器切换为休眠模式或关机状态,降低能源消耗。金顺福等^[7]针对异构用户的服务质量需求及云数据中心能源浪费问题,提出一种融合区分服务和速率调整的请求调度策略。上述策略专注于降低系统能耗的云资源管理方案,而忽略了云用户对响应性能的需求。

本文兼顾云用户的服务质量与云系统的节能水平,引入休眠延迟机制与休眠唤醒阈值,提出一种基于新型休眠机制的云任务调度策略。考虑物理机的唤醒过程,建立连续时间排队模型,研究云用户的随机行为。结合云系统中任务总数与物理机状态,构建二维连续时间 Markov 链,进行云任务平均逗留时间与云系统节能水平等性能指标的理论分析。按照

① 国家自然科学基金(61872311, 61472342)和河北省自然科学基金(F2017203141)资助项目。

② 女,1992 年生,硕士生;研究方向:计算机通信网络的系统建模与性能分析;联系人, E-mail: ysuwxs@163.com

③ 通信作者, E-mail: jsf@ysu.edu.cn
(收稿日期:2018-05-11)

云任务调度策略的工作原理进行仿真实验,给出相关性能指标的统计结果。权衡云用户服务质量与云系统节能水平,构造系统成本函数。通过动态改变交叉概率与变异概率,改进遗传算法,以最小化系统成本为目标,在不同休眠唤醒阈值下给出休眠参数与休眠延迟参数的联合优化方案。

1 云任务调度策略及系统模型

1.1 云任务调度策略

为提高并行处理能力,多个虚拟机可通过虚拟化技术同时托管在一台物理机上。在传统的云任务调度策略中,分布在一个或多个物理机上的虚拟机始终保持活跃,造成巨大的能耗浪费。

置空闲的物理机于休眠模式可有效减少能源消耗。但是,休眠往往会降低系统响应性能。基于此,兼顾云用户的响应性能与云系统的节能水平,提出带有休眠延迟机制与休眠唤醒阈值的云任务调度策略。考虑休眠模式引起的负面影响,从保证云用户的服务质量出发,设定一个休眠延迟定时器。当系统变空时,物理机在休眠延迟期内保持活跃状态。在休眠延迟期内到达的云任务可立即得到服务,从而保证云用户的响应性能。如果在休眠延迟期内没有云任务到达,物理机将进入周期性休眠状态,多个休眠周期构成一个休眠期。考虑物理机由休眠状态切换至活跃状态的唤醒过程会导致额外的能量消耗,设定一个休眠唤醒阈值 $N(N > 0)$ 。在一个休眠周期结束时刻,如果系统缓冲区中等待的云任务数量未达到休眠唤醒阈值 N ,便开启另一个休眠周期,即物理机继续处于休眠状态。否则,物理机经历一个唤醒过程后恢复到活跃状态,云任务便分配在不同虚拟机上以获得服务,从而提高云系统节能水平。

在所提出的云任务调度策略中,物理机共有 4 种状态:活跃状态,休眠延迟状态,休眠状态与唤醒状态。物理机的状态转移过程如图 1 所示。

(1) 活跃状态 在此状态下的物理机及物理机上的虚拟机全部保持活跃,且至少有一个虚拟机正在处理云任务。

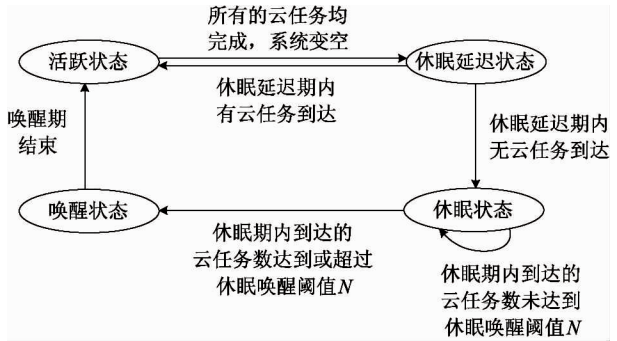


图 1 物理机状态转移图

(2) 休眠延迟状态 当系统变空,即所有虚拟机空闲时,物理机将进入休眠延迟状态,同时开启休眠延迟定时器。如果在休眠延迟期内有云任务到达系统,物理机将由休眠延迟状态返回到活跃状态,其中的一个虚拟机立即开始为新到达的云任务服务。否则,物理机将在休眠延迟期结束时刻,由休眠延迟状态切换到周期性休眠状态。

(3) 休眠状态 当物理机进入周期性休眠状态时,立即开启休眠定时器。休眠状态下,内存中的数据保存至硬盘上的一个休眠文件中,物理机将切断除内存外其他配件的电源,不再为系统中的云任务提供服务。休眠期内到达的云任务在系统缓冲区中排队等待。在一个休眠周期的结束时刻,如果系统缓冲区内的云任务数量少于休眠唤醒阈值 N ,物理机将进入另一个休眠周期,并重新启动休眠定时器。否则,物理机结束休眠,进入唤醒状态。

(4) 唤醒状态 当物理机处于唤醒状态时,系统会将休眠文件中的内容直接读入内存。唤醒期内到达的云任务继续在系统缓冲区中排队等待。唤醒期结束后,即物理机恢复到活跃状态时,按照先到先服务的原则,为云任务提供服务。

1.2 系统模型

将云计算系统中的云任务视为顾客,每个虚拟机视为服务器,活跃状态视为忙期,休眠状态视为休假,休眠延迟状态视为休假延迟,唤醒状态视为启动过程,休眠唤醒阈值视为 N -策略,建立一个带有休假延迟与启动过程的同步 N -策略多重休假排队模型。

假设云任务到达服从参数为 $\lambda(0 < \lambda < \infty)$ 的 Poisson 过程;虚拟机对一个云任务的处理时间服从

参数为 $\mu (0 < \mu < \infty)$ 的指数分布;休眠定时器与休眠延迟定时器的时长分别服从参数为 $\theta (0 < \theta < \infty)$ 与 $\beta (0 < \beta < \infty)$ 的指数分布;物理机的唤醒时间服从参数为 $\alpha (0 < \alpha < \infty)$ 的指数分布。假设云任务的到达间隔,服务时间,休眠时间,休眠延迟时间与唤醒时间相互独立。假设系统中虚拟机的数量为 k ,系统缓冲区大小无限。

系统中有 k 个虚拟机, $\lambda (0 < \lambda < \infty)$ 为云任务的到达率, $\mu (0 < \mu < \infty)$ 为一个虚拟机的服务率。只有在云任务的到达率 λ 小于系统服务率 $k\mu$ 的条件下,系统才能达到稳定状态。定义 $\rho = \frac{\lambda}{k\mu}$ 为系统服务强度,系统稳态的充分必要条件为 $\frac{\lambda}{k\mu} < 1$ 。

令随机变量 $X(t) = i, i = 0, 1, \dots$ 表示 t 时刻系统中云任务的数量,称为系统水平。令随机变量 $Y(t) = j, j = 0, 1, 2$ 表示物理机所处的状态: $j = 0$ 表示物理机处于休眠状态,当 $i = 0$ 时, $j = 1$ 表示物理机处于休眠延迟状态,当 $i \geq 1$ 时, $j = 1$ 表示物理机处于唤醒状态, $j = 2$ 表示物理机处于活跃状态。 $\{(X(t), Y(t)), t \geq 0\}$ 构成一个二维的连续时间 Markov 链。

令 $\pi_{i,j}$ 表示稳态下系统水平为 i 且物理机状态为 j 的概率分布。 $\pi_{i,j}$ 定义为

$$\pi_{i,j} = \lim_{t \rightarrow \infty} P\{X(t) = i, Y(t) = j\}, (i, j) \in \Omega$$

令 $\pi_i (i \geq 0)$ 表示系统水平为 i 的稳态概率向量,则 $\pi_i (i \geq 0)$ 形式为

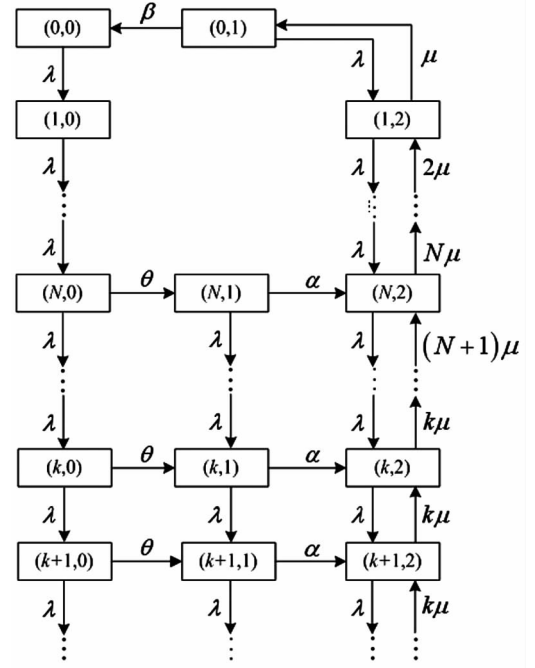
$$\pi_i = \begin{cases} (\pi_{0,0}, \pi_{0,1}), & i = 0 \\ (\pi_{i,0}, \pi_{i,1}, \pi_{i,2}), & i = 1, 2, \dots \end{cases}$$

2 系统模型的稳态分析

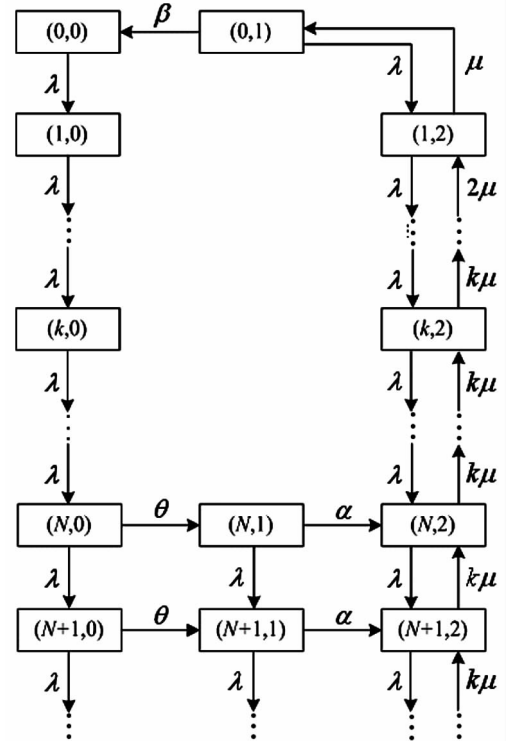
2.1 状态转移

根据系统中虚拟机的数量 k 与休眠唤醒阈值 N 之间的关系,Markov 链 $\{(X(t), Y(t)), t \geq 0\}$ 存在 2 种形式的状态转移,如图 2 所示。

由状态转移图的结构可知,状态转移只发生在相邻的系统水平之间,表明二维 Markov 链 $\{(X(t), Y(t)), t \geq 0\}$ 是一种拟生灭过程。



(a) $N \leq k$



(b) $N > k$

图 2 Markov 链的状态转移

2.2 稳态分布

由图 2 可知,当 $N \leq k$ 时,状态转移从水平 k 开始重复,当 $N > k$ 时,状态转移从水平 N 开始重复。

使用矩阵几何解方法^[8]分析上述拟生灭过程,转移率矩阵 $\mathbf{R}$ 形式为

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ 0 & r_{22} & r_{23} \\ 0 & 0 & r_{33} \end{pmatrix}$$

建立二次代数方程组:

$$\begin{cases} \lambda - r_{11}(\lambda + \theta) = 0 \\ r_{11}\theta - r_{12}(\lambda + \alpha) = 0 \\ k\mu(r_{11}r_{13} + r_{12}r_{23} + r_{13}r_{33}) + r_{12}\alpha - r_{13}(\lambda + k\mu) = 0 \\ \lambda - r_{22}(\lambda + \alpha) = 0 \\ k\mu(r_{22}r_{23} + r_{23}r_{33}) + r_{22}\alpha - r_{23}(\lambda + k\mu) = 0 \\ \lambda + r_{33}^2k\mu - r_{33}(\lambda + k\mu) = 0 \end{cases}$$

(1)

在 $\rho = \frac{\lambda}{k\mu} < 1$ 的条件下,求解方程组(1),得到的转移率矩阵 $\mathbf{R}$ 为

$$\mathbf{R} = \begin{pmatrix} \frac{\lambda}{\lambda + \theta} & \frac{\lambda\theta}{(\lambda + \theta)(\lambda + \alpha)} & \rho \\ 0 & \frac{\lambda}{\lambda + \alpha} & \rho \\ 0 & 0 & \rho \end{pmatrix}$$

根据转移率矩阵 $\mathbf{R}$ 的解析结果,易知 $SP(\mathbf{R}) = \max\{r_{11}, r_{22}, r_{33}\} < 1$. 故二维 Markov 链 $\{(X(t), Y(t)), t \geq 0\}$ 正常返,存在平稳概率分布. 结合高斯-塞德尔方法与矩阵几何解方法,可以给出稳态概率分布 $\pi_i (i = 0, 1, \cdots)$ 的数值解.

3 系统性能指标与系统实验

3.1 系统性能指标

云任务平均逗留时间 W 定义为云任务从进入系统开始到服务完成为止的平均时间长度. 基于系统模型的稳态分布,利用 Little 公式^[9],云任务平均逗留时间 W 的表达式为

$$W = \frac{1}{\lambda} \left(\sum_{i=0}^{\infty} i(\pi_{i,0} + \pi_{i,1} + \pi_{i,2}) \right)$$

(2)

云系统节能水平定义为单位时间内物理机所减少的能量消耗. 活跃状态与休眠延迟状态正常耗能,休眠状态耗能减少,唤醒状态耗能高于活跃状态耗能. 此外,在每个休眠周期结束时刻,物理机的监

听过程会增加额外的能量消耗. 令 C_a 为在活跃状态与休眠延迟状态的能耗水平, C_s 表示在休眠状态的能耗水平, C_l 是在唤醒状态的能耗水平, C_l 表示单次监听引起的能量消耗. 云系统节能水平 S 的表达式为

$$S = \sum_{i=0}^{\infty} \pi_{i,0} \times (C_a - C_s) - \sum_{i=N}^{\infty} \pi_{i,1} \times C_l$$
$$- \sum_{i=0}^{\infty} \pi_{i,0} \times \theta \times C_l$$

(3)

3.2 系统实验

为量化休眠参数,休眠延迟参数与休眠唤醒阈值对云任务调度策略的影响,进行性能指标变化趋势的系统实验.

实验的硬件环境为 Intel(R) Core(TM), i7-4790 CPU @ 3.60 GHz, 8.00 GB RAM. 交互式软件的开发环境为 MATLAB R2011a. 在保证系统稳态,即 $\rho = \frac{\lambda}{k\mu} < 1$ 的前提下,设置如表 1 所示的实验参数,进行系统实验,揭示云任务平均逗留时间与云系统节能水平的变化规律.

表 1 实验参数设置

参数	值
一个物理机中虚拟机总数 k	20 台
活跃虚拟机服务率 μ	0.2 个/ms
云任务到达率 λ	0.4 个/ms
唤醒过程时间参数 α	0.1 次/ms
活跃物理机耗能水平 C_a	20 mW
休眠物理机耗能水平 C_s	2 mW
物理机唤醒过程耗能水平 C_l	12 mW
物理机监听过程耗能 C_l	4 mJ/次

图 3 刻画了休眠参数 θ , 休眠延迟参数 β 与休眠唤醒阈值 N 等参数对云任务的平均逗留时间 W 的影响.

固定休眠唤醒阈值与休眠延迟参数,云任务的平均逗留时间随着休眠参数的增大呈下降趋势. 休眠参数越大,在一个休眠周期内到达的任务需要等待休眠周期结束的时间越短,因而云任务的平均逗留时间减少.

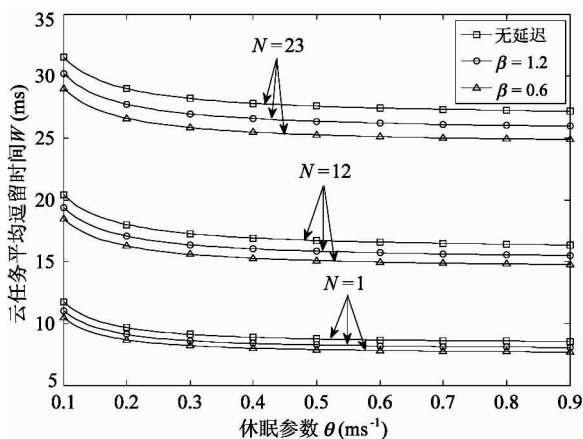


图3 云任务平均响应时间的变化趋势

固定休眠唤醒阈值与休眠参数,云任务的平均逗留时间随着休眠延迟参数的增大呈上升趋势。休眠延迟参数越大,休眠延迟期越短,期间到达的云任务立即得到服务的可能性越小,因而云任务的平均逗留时间增加。

固定休眠延迟参数与休眠参数,云任务的平均逗留时间随着休眠唤醒阈值的增大呈上升趋势。休眠唤醒阈值越大,等待云任务数达到休眠唤醒阈值的时间越长,即休眠状态的持续时间越长,因而云任务的平均逗留时间增加。

图4刻画了休眠参数 θ ,休眠延迟参数 β 与休眠唤醒阈值 N 等参数对云系统节能水平 S 的影响。

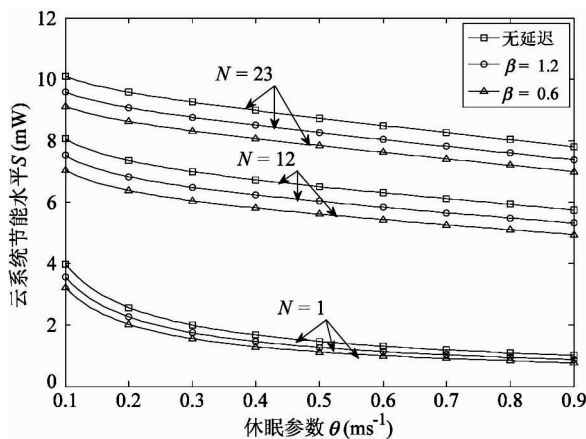


图4 云系统节能水平的变化趋势

固定休眠唤醒阈值与休眠延迟参数,云系统节能水平随着休眠参数的增大呈下降趋势。休眠参数越大,一个休眠周期持续时间越短,则监听越频繁,

由监听引起的额外能耗越多,因而云系统的节能水平减少。

固定休眠唤醒阈值与休眠参数,云系统的节能水平随着休眠延迟参数的增大呈上升趋势。休眠延迟参数越大,休眠延迟期持续时间越短,因而物理机将更早地进入休眠状态,引起云系统节能水平的增加。

固定休眠延迟参数与休眠参数,云系统的节能水平随着休眠唤醒阈值的增大呈上升趋势。休眠唤醒阈值越大,休眠期内云任务数达到休眠唤醒阈值的时间越长,即休眠状态的持续时间越长,因而云系统的节能水平增加。

以上的实验结果表明,休眠参数,休眠延迟参数与休眠唤醒阈值对系统性能有着不容忽视的影响。对具有高响应性能需求的云任务来说,休眠唤醒阈值与休眠延迟参数越小越好,而休眠参数越大越好。对具有高效节能需求的云系统来说,休眠唤醒阈值与休眠延迟参数越大越好,而休眠参数越小越好。因此,结合云任务与云系统的不同要求,需要在不同休眠唤醒阈值下对休眠延迟参数与休眠参数进行联合优化。

4 系统优化

为探究云任务平均逗留时间与云系统节能水平之间的折衷关系,需要构造系统成本函数。由于云任务平均逗留时间的量纲与云系统节能水平的量纲不同,需要对性能指标进行归一化处理。在休眠参数范围在 $[0.1, 0.9]$ 时,云任务平均逗留时间的值域为 $[7.7042, 31.5460]$,云系统节能水平的值域为 $[0.7706, 10.0967]$ 。采用 $\min - \max$ 标准化方法,建立映射 $x \rightarrow \frac{x - \min}{\max - \min}$,归一化云任务平均逗留时间 W 与云系统节能水平 S 。其中, $\min$ 为值域的最小值, $\max$ 为值域的最大值。系统成本函数 F 为

$$F = f_1 \times \frac{W - 7.7042}{23.8418} - f_2 \times \frac{S - 0.7706}{9.3261} \quad (4)$$

其中, f_1 表示云任务平均逗留时间 W 对系统成本的影响因子, f_2 表示云系统节能水平 S 对系统成本的影响因子。在实际应用中,对于高响应需求的云系

统来说,参数 f_2 的值就要设置的大些;而对于高节能需求的云系统来说,参数 f_1 的值则要设置的大些。

由于云任务的平均逗留时间与云系统的节能水平的精确表达式很难给出,系统成本函数的严格单调性无法确定。本文借助于智能优化算法^[10],联合优化休眠参数与休眠延迟参数。

在传统遗传算法中,交叉概率与变异概率往往是静态设置的,算法容易早熟并陷入局部最优。调整交叉概率与变异概率,使其随种群结构的特征动态变化,是提高遗传算法运行效率、克服早熟现象、避免局部收敛问题的有效手段。基于线性回归的方法^[11],令交叉概率随进化代数的增加而增加,变异概率随进化代数的增加而减小,提出一种改进的遗传算法。改进的遗传算法的主要步骤如下。

步骤 1 初始化休眠参数上限 θ_u 与下限 θ_l , 休眠延迟参数上限 β_u 与下限 β_l , 种群规模 M ,最大交叉概率 P_{cm} 与最小交叉概率 P_{cl} , 最大变异概率 P_{mm} 与最小变异概率 P_{ml} 。初始迭代次数为 $gen = 1$, 最大迭次数 $gen_{max} = 50$ 。

步骤 2 在休眠参数与休眠延迟参数的搜索空间范围内,初始化种群 X 中各成员 $(\theta, \beta)_i, i \in \{1, 2, \cdots, M\}$ 。

步骤 3 计算每个个体的适应度 $F((\theta, \beta)_i)$, 累积概率 $P((\theta, \beta)_i)$ 与选择概率 $C((\theta, \beta)_i)$ 。

$$F((\theta, \beta)_i) = f_1 \times \frac{W((\theta, \beta)_i) - 7.7042}{23.8418} - f_2 \times \frac{S((\theta, \beta)_i) - 0.7706}{9.3261}$$

$$P((\theta, \beta)_i) = \frac{F((\theta, \beta)_i)}{\sum_{i=1}^M F((\theta, \beta)_i)}$$
$$C((\theta, \beta)_i) = \sum_{j=1}^i P((\theta, \beta)_j)$$

其中, $W(\theta, \beta)_i$ 与 $S(\theta, \beta)_i$ 分别表示休眠参数与休眠延迟参数组合为 $(\theta, \beta)_i$ 时云任务的平均逗留时间与云系统的节能水平。

步骤 4 计算交叉概率 P_c 与变异概率 P_m 。

$$P_c = P_{cl} - \frac{(P_{cm} - P_{cl}) \times gen}{gen_{max}}$$

$$P_m = P_{ml} + \frac{(P_{mm} - P_{ml}) \times gen}{gen_{max}}$$

步骤 5 执行遗传操作,更新种群 X 。

for $i = 1:M$

$$slen = selection(X, C_i)$$

% 选择两个个体进行交叉与变异

$$X = crossover(X, slen, P_c)$$

% 将选出的两个个体进行交叉

$$X = mutation(X, slen, P_m)$$

% 将交叉后的两个个体进行变异

endfor

步骤 6 检查迭代次数。

if $gen < gen_{max}$

then $gen = gen + 1$, 跳转到步骤 3

endif

步骤 7 找出种群 X 中使系统成本达到最小的个体。

$$(\theta, \beta)^* = \underset{i \in \{1, 2, \cdots, M\}}{\operatorname{argmin}} \{F((\theta, \beta)_i)\}$$
$$F((\theta, \beta)^*) = f_1 \times \frac{W((\theta, \beta)^*) - 7.7042}{23.8418} - f_2 \times \frac{S((\theta, \beta)^*) - 0.7706}{9.3261}$$

步骤 8 输出最优参数组合 $(\theta, \beta)^*$ 与最小系统成本 $F((\theta, \beta)^*)$ 。

利用改进的遗传算法,数值求解不同休眠唤醒阈值下的最小系统成本。设定系统成本影响因子 $f_1 = 1.6, f_2 = 0.9$, 给出休眠参数与休眠延迟参数的优化组合。优化结果如表 2 所示。

表 2 休眠参数与休眠延迟参数的联合优化结果

休眠唤醒 阈值 N	休眠参数与休眠延迟参数 的最优组合 $(\theta, \beta)^*$	最小系统 成本 F^*
10	(0.1345, 0.7136)	0.2956
16	(0.3752, 0.5488)	0.5919
22	(0.7732, 0.2926)	0.9221

联合优化结果是通过最小化系统成本得出的。表 2 中休眠参数与休眠延迟参数的联合优化结果是基于表 1 中设定的实验参数得到的。若采用另外一组实验参数,遗传算法会对应得到不同的联合优化

结果。

利用相同的实验参数、相同的随机初始化条件,分别执行改进的遗传算法与传统的遗传算法。改进的遗传算法运行时间的平均值约为 65 s,传统的遗传算法运行时间的平均值约为 72 s。从时间数值对比上看,改进的遗传算法具有更快的寻优速度,表明动态调整交叉概率与变异概率是有效的。

5 结 论

兼顾云任务高响应性能需求与云系统高效节能需求,引入新型休眠机制,提出了一种云任务调度策略。通过建立带有休假延迟与启动过程的同步 N -策略多重休假排队模型,给出了云任务平均逗留时间与云系统节能水平等性能指标。实验结果表明,通过设置合理的系统参数,所提策略在保证用户响应性能的前提下可以有效降低系统能量消耗。折衷不同性能指标,从经济学的角度出发,构建了系统成本函数。改进遗传算法,给出休眠参数与休眠延迟参数的联合优化方案,实现了系统成本最小化。

参考文献

- [1] Alsarhan A, Itradat A, Aldubai A Y, et al. Adaptive resource allocation and provisioning in multi-service cloud environments[J]. *IEEE Transactions on Parallel & Distributed Systems*, 2018, 29(1): 31-42
- [2] Hasan M S, Alvares F, Ledoux T, et al. Investigating energy consumption and performance trade-off for interactive cloud application[J]. *IEEE Transactions on Sustainable Computing*, 2017, 2(2): 113-126
- [3] Vakiliinia S, Heidarpour B, Cheriet M. Energy efficient resource allocation in cloud computing environments[J]. *IEEE Access*, 2017, 4(99): 8544-8557
- [4] Yang J, Jiang B, Lv Z, et al. A task scheduling algo-

rithm considering game theory designed for energy management in cloud computing[EB/OL]. <https://doi.org/10.1016/j.future.2017.03.024>. htm; Future Generation Computer Systems, 2017

- [5] Jin S, Hao S, Yue W. Energy-efficient strategy with a speed switch and a multiple-sleep mode in cloud data centers[C]. In: Proceedings of the 12th International Conference on Queueing Theory & Network Applications, Qinhuangdao, China, 2017. 143-154
- [6] Guo L, Zhang Y, Zhao S. Heuristic algorithms for energy and performance dynamic optimization in cloud computing[J]. *International Computing and Informatics*, 2017, 36(6): 1335-1360
- [7] 金顺福, 薛元铮, 顾蕊. 融合区分服务和速率调整的请求调度策略[J]. 高技术通讯, 2017, 27(9-10): 790-799
- [8] 田乃硕. 休假随机服务系统[M]. 北京:北京大学出版社, 2001. 263-292
- [9] Wang B, Li J. Load balancing task scheduling based on multi-population genetic algorithm in cloud computing[C]. In: Proceedings of the 35th Chinese Control Conference, Chengdu, China, 2016. 27-29
- [10] Yue D, Yu J, Yue W. A Markovian queue with two heterogeneous servers and multiple vacations[J]. *Journal of Industrial & Management Optimization*, 2017, 5(3): 453-465
- [11] Makasarwala H A, Hazari P. Using genetic algorithm for load balancing in cloud computing[C]. In: Proceedings of the 8th International Conference on Electronics, Computers and Artificial Intelligence, Ploiesti, Romania, 2017. 1-6

Research on a cloud task scheduling strategy based on a novel sleep mechanism

Wang Xiushuang^{* ** ***}, Jin Shunfu^{* ** ***}

(^{*} School of Information Science and Engineering, Yanshan University, Qinhuangdao 066004)

(^{**} Key Laboratory for Computer Virtual Technology and System Integration of Hebei Province, Qinhuangdao 066004)

(^{***} Key Laboratory for Software Engineering of Hebei Province, Qinhuangdao 066004)

Abstract

In order to improve the energy conservation level of cloud system and satisfy the response performance of cloud users, a cloud task scheduling strategy and its system performance optimization are investigated. By introducing a sleep-delay mechanism and a waking-up threshold, a cloud task scheduling strategy based on a novel sleep mechanism is proposed. According to the proposed cloud task scheduling strategy and the stochastic behavior of cloud users, considering the wake-up procedure of the physical machine, a synchronous N -policy multiple vacation queueing model with vacation-delay and set-up is constructed. From the view of the average sojourn time of cloud tasks and the energy conservation level of cloud system, the system performance of the cloud task scheduling strategy is evaluated by using the quasi-birth-and-death processes and the matrix-geometric solution. Numerical experiments are provided to validate the effectiveness of the proposed strategy and the rationality of the model analysis, and to reveal the trade-off between the response performance and energy conservation. By dynamically changing the cross-over probability and the mutation probability in the genetic algorithm, the cloud task scheduling strategy is optimized to minimize the system cost.

Key words: cloud computing, task scheduling, sleep-delay, waking-up threshold, queueing model, set-up, intelligent optimization