Statistics and Application 统计学与应用, 2024, 13(1), 189-201
Published Online February 2024 in Hans. https://www.hanspub.org/journal/sa
https://doi.org/10.12677/sa.2024.131021

Hans汉斯

带优先权用户和半休眠机制的云系统节能分析

张丽丽，姚懿恬

燕山大学理学院，河北 秦皇岛

收稿日期：2024年1月12日；录用日期：2024年2月22日；发布日期：2024年2月29日

摘　要

针对云用户呈指数增长现状，云环境下为了在保证用户服务质量的同时降低系统能耗、提高系统收益的问题。提出运用带抢占优先权和服务台同步工作休假的M/M/c排队模型对云系统性能和系统收益进行建模分析。建立三维Markov过程，得到其无穷小生成元矩阵，采用拟生灭过程理论求出系统的稳态分布，得出两类顾客的平均队长、内存溢出率、系统总能耗等稳态性能指标，并建立相应的系统收益函数。通过数值模拟探讨参数对系统稳态性能指标的影响，研究VM半休眠机制对系统总能耗的影响以及参数对系统收益函数的作用规律。在保证两类顾客服务质量的前提下，通过动态调整VM数、内存空间和服务率来降低系统总能耗、增加系统收益。

关键词

排队论，抢占优先权，同步半休眠，系统总能耗，系统收益

Energy Saving Analysis of Cloud System with Priority Users and Semi-Sleep Mechanism

Lili Zhang, Yitian Yao

School of Science, Yanshan University, Qinhuangdao Hebei

Received: Jan. 12th, 2024; accepted: Feb. 22nd, 2024; published: Feb. 29th, 2024

Abstract

In order to reduce system energy consumption and improve system profit while ensuring user service quality, this paper proposes to model and analyze the cloud system performance based on the M/M/c queue with preemptive priority and server synchronous vacation. The three-dimensional Markov process is established, its infinitesimal generator matrix is gained, the steady-state distribution of the system is gained by using the theory of quasi birth-and-death, the steady-state performance indicators such as the average queue length of the two types of customers, memory overflow rate, and total energy consumption of the system are gained, and the corresponding system

文章引用: 张丽丽, 姚懿恬. 带优先权用户和半休眠机制的云系统节能分析[J]. 统计学与应用, 2024, 13(1): 189-201.
DOI: 10.12677/sa.2024.131021

benefit function is established. Through numerical simulation, the effect of parameters on the steady-state performance index and profit of the system is discussed, and the effect of the semi-hibernation mechanism of virtual machine on the total energy consumption of the system are studied. Under the premise of ensuring the service quality of customer service of the two types, it is to reduce the total energy consumption of the system and to improve system benefit by dynamically adjusting the number of virtual machines, memory space and service rate.

Keywords

Queuing Theory, Preemption, Synchronous Half-Sleep, Total System Energy Consumption, System Income

1. 引言

云计算通过互联网为远程用户提供可靠性高且廉价的计算资源，云计算中心通常会根据不同的业务需求来配备不同的服务模式进行数据传输和任务处理，例如一些用户需要在特定的时间内使用资源，而其他用户则对时间的要求不严格。根据云计算中心的服务等级协议(SLA)给用户赋予优先级，以保证高优先级用户的服务质量(QoS)。WHITE [1]最早提出了抢占优先权排队系统。RAO [2]研究了在非抢占优先权下的(M1 + M2)/G/1 排队模型，得出稳定性条件。GAIL 等[3]研究了具有多个服务台和两类优先权顾客的排队模型，并给出了两类顾客的分布表达式。徐秀丽[4]研究了带可变服务率和启动期的 M/M/1 休假排队模型。徐秀丽[5]研究了带(e, d)休假策略的 M/M/c 排队模型。郭闪闪[6]研究了带抢占优先权的排队系统，给出顾客的平均逗留时间等性能指标，建立收益函数并通过数值试验寻找最优参数。SI [7]等研究了具有不耐烦顾客、服务台故障可修的 M/M/c 排队模型。

在互联网和信息技术不断进步的今天，各行业的计算资源都开始转移到云计算中心，用户对云计算中心性能需求的增长与系统能耗成本之间的矛盾日益凸显。为了满足用户的请求并能够获得最大的利润，许多学者将休假排队理论运用到云资源节能策略中。SAHOO 和 GOSWAMI [8]提出了基于休假排队的虚拟机(VM)唤醒和休假策略。李吉良等[9]研究了半休眠模式下带唤醒阈值的 VM 调度策略，通过数值分析评估 VM 调度策略的系统性能。金顺福[10]等研究了基于唤醒阈值和休眠定时器双重控制的 VM 分簇调度策略，建立了带有(N,T)策略的多重休假排队模型，有效节约系统能耗。GUO 等[11]研究了云任务随机到达且带有任务延时的 VM 调度问题。MA 等[12]研究了同步休眠模式和异步休眠模式相结合的 VM 调度策略。JIN 等[13] [14]研究了具有多种睡眠机制的 VM 调度策略以及带有唤醒阈值的集群 VM 分配策略，建立带有 N 策略和部分服务台异步休假的 M/M/c 排队模型进行建模分析。LI 和 JIN [15] [16]研究了带有抢占优先权用户的多服务台排队模型和带有异构边缘 MEC 系统的云任务卸载策略。CUI 等[17]研究了基于半休眠机制和可变服务率的云系统动态节能策略。

在上述研究的基础上，考虑到云计算中心为了根据不同的业务需求而提供不同的服务，为了保证高优先级顾客的 Qos，引入优先权策略。同时引入 c 台 VM 同步半休眠策略以减少系统空闲时的能耗。最后考虑到 VM 的内存空间是有限的，过多的用户请求会导致内存溢出，因此设低优先级顾客的等待空间是有限的。建立一个带抢占优先权和服务台同步工作休假的 M/M/c 排队模型，推导出两类顾客的平均队长，内存溢出率、系统总能耗等稳态性能指标，并建立相应的系统收益函数，分析参数对稳态性能指标和系统收益的影响。

2. 模型描述

2.1. 用户到达规则

针对云计算中心的运行机制，将用户每一次对服务器进行访问的行为当作一次任务到达过程，根据云计算中心的 SLA，本文将用户分为高优先级顾客(第一类顾客)和低优先级顾客(第二类顾客)。模型中假定两类顾客分别在两个缓冲器中各自排队，并设两类顾客的到达时间间隔分别服从参数为 λ_1 和 λ_2 的负指数分布。考虑到 VM 的内存空间是有限的，过多的用户请求会导致内存溢出，因此设第二类顾客的等待空间 $N(N>c)$ 是有限的。

2.2. 队列优先权控制规则

为了保证高优先级顾客的 QoS,设第一类顾客具有抢占优先权,抢占规则为当第一类顾客到达系统后，如果系统中无空闲 VM 则抢占正在为第二类顾客服务的 VM，第二类顾客将回到自己队列的首部位置重新等待服务。设两类顾客的服务时间分别服从参数为 μ_1 和 μ_2 的负指数分布。

2.3. 虚拟机休眠规则

针对虚拟机休眠策略，本文引入 c 台 VM 同步半休眠策略。即当 c 个 VM 均空闲时，系统将进入半休眠状态。若在半休眠状态内有新顾客的请求，则 VM 就会以较低的服务率来服务顾客，待半休眠状态结束时变成以正常的服务率来服务顾客；若半休眠状态结束时系统中仍无新顾客的请求，则进入下一个半休眠状态。设系统休眠长度服从参数为 ξ 的负指数分布，两类顾客在半休眠状态的服务时间分别服从参数为 $v_1(v_1<\mu_1)$ 和 $v_2(v_2<\mu_2)$ 的负指数分布。假设系统中两类顾客的到达时间、服务时间均相互独立，且均服从先到先服务 FCFS 的服务规则。

3. 模型建立与分析

3.1. 拟生灭过程

根据模型描述建立具有抢占优先权的 M/M/c 工作休假排队系统，设 $N_1(t)$ 和 $N_2(t)$ 分别表示高优先级顾客和低优先级顾客在时刻 t 的请求数。令

$$S(t)=\begin{cases}0, & \text{时刻 } t \text{ 系统处于半休眠状态},\\ 1, & \text{时刻 } t \text{ 系统处于正常工作状态}.\end{cases}$$

则 $\{(N_1(t),N_2(t),S(t)),t\geq 0\}$ 构成一个三维拟生灭过程，其状态空间为

$$\Omega=\{(0,0,0)\}\cup\{(0,j,s),1\leq j\leq N,s=0,1\}\cup\{(i,j,s),i\geq 1,0\leq j\leq N,s=0,1\}.$$

考虑高优先级顾客，其状态 0 包含如下水平：$(0,0,0),(0,1,0),(0,1,1),\cdots,(0,N,0),(0,N,1)$，状态 i 包含如下水平：$(i,0,0),(i,0,1),(i,1,0),(i,1,1),\cdots,(i,N,0),(0,N,1),i\geq 1$，将状态按字典序排列，无穷小生成元 $\boldsymbol{Q}$ 如下

$$\boldsymbol{Q}=\begin{bmatrix}
\boldsymbol{A}_0 & \boldsymbol{C}_0 & & & & & & & & \\
\boldsymbol{B}_1 & \boldsymbol{A}_1 & \boldsymbol{C}_1 & & & & & & & \\
 & \boldsymbol{B}_2 & \boldsymbol{A}_2 & \boldsymbol{C}_1 & & & & & & \\
 & & \boldsymbol{B}_3 & \boldsymbol{A}_3 & \boldsymbol{C}_1 & & & & & \\
 & & & \ddots & \ddots & \ddots & & & & \\
 & & & & \boldsymbol{B}_{c-1} & \boldsymbol{A}_{c-1} & \boldsymbol{C}_1 & & & \\
 & & & & & \boldsymbol{B}_c & \boldsymbol{A}_c & \boldsymbol{C}_1 & & \\
 & & & & & & \boldsymbol{B}_c & \boldsymbol{A}_c & \boldsymbol{C}_1 & \\
 & & & & & & & \ddots & \ddots & \ddots
\end{bmatrix},$$

其中 $\boldsymbol{A}_0$ 为 $2N+1$ 阶方阵， $\boldsymbol{A}_i, 1\le i\le c$ 为 $2N+2$ 阶方阵， $\boldsymbol{C}_0$ 为 $(2N+1)(2N+2)$ 阶矩阵， $\boldsymbol{C}_1$ 为 $2N+2$ 阶方阵， $\boldsymbol{B}_1$ 为 $(2N+2)(2N+1)$ 阶矩阵， $\boldsymbol{B}_i, 2\le i\le c$ 为 $2N+2$ 阶方阵。且

$$\boldsymbol{C}_0=\begin{bmatrix}\lambda_1 & 0 & & & & \\ & 0 & \lambda_1 & & & \\ & & 0 & \lambda_1 & & \\ & & & \ddots & \ddots & \\ & & & & 0 & \lambda_1\end{bmatrix},\boldsymbol{C}_1=\begin{bmatrix}\lambda_1 & & & & \\ & \lambda_1 & & & \\ & & \lambda_1 & & \\ & & & \ddots & \\ & & & & \lambda_1\end{bmatrix},$$

$$\boldsymbol{B}_1=\begin{bmatrix}\nu_1 & & & & & \\ \mu_1 & 0 & & & & \\ & \nu_1 & 0 & & & \\ & & \mu_1 & 0 & & \\ & & & \ddots & \ddots & \\ & & & & \nu_1 & 0 \\ & & & & & \mu_1\end{bmatrix},$$

$$\boldsymbol{B}_i=\begin{bmatrix}i\nu_1 & & & & & \\ & i\mu_1 & & & & \\ & & i\nu_1 & & & \\ & & & i\mu_1 & & \\ & & & & \ddots & \\ & & & & & i\nu_1 \\ & & & & & & i\mu_1\end{bmatrix},\ 2\le i\le c-1$$

$$\boldsymbol{B}_c=\begin{bmatrix}c\nu_1 & & & & & \\ & c\mu_1 & & & & \\ & & c\nu_1 & & & \\ & & & c\mu_1 & & \\ & & & & \ddots & \\ & & & & & c\nu_1 \\ & & & & & & c\mu_1\end{bmatrix},$$

$$\boldsymbol{A}_0=\begin{bmatrix}-(\lambda_1+\lambda_2) & \lambda_2 & & & & & & & & & & \\ \nu_2 & \delta_1 & \xi & \lambda_2 & & & & & & & & \\ \mu_2 & 0 & \omega_1 & 0 & \lambda_2 & & & & & & & \\ & 2\nu_2 & 0 & \delta_2 & \xi & \lambda_2 & & & & & & \\ & & 2\mu_2 & 0 & \omega_2 & 0 & \lambda_2 & & & & & \\ & & & & \ddots & & \ddots & & & & & \\ & & & & c\nu_2 & 0 & \delta_c & \xi & \lambda_2 & & & \\ & & & & & c\mu_2 & 0 & \omega_c & 0 & \lambda_2 & & \\ & & & & & & c\nu_2 & 0 & \delta_c & \xi & \lambda_2 & \\ & & & & & & & c\mu_2 & 0 & \omega_c & 0 & \lambda_2 \\ & & & & & & & & & \ddots & & \ddots \\ & & & & & & & & & c\nu_2 & 0 & \delta_N & \xi \\ & & & & & & & & & & c\mu_2 & 0 & \omega_N\end{bmatrix},$$

其中

$$\delta_i=\begin{cases}-\left(\lambda_1+\lambda_2+\xi+iv_2\right), & 1\le i<c,\\ -\left(\lambda_1+\lambda_2+\xi+cv_2\right), & c\le i<N,\\ -\left(\lambda_1+\xi+cv_2\right), & i=N,\end{cases}$$

$$\omega_i=\begin{cases}-\left(\lambda_1+\lambda_2+i\mu_2\right), & 1\le i<c,\\ -\left(\lambda_1+\lambda_2+c\mu_2\right), & c\le i<N,\\ -\left(\lambda_1+c\mu_2\right), & i=N.\end{cases}$$

$$\boldsymbol{A}_i=\begin{bmatrix}
\delta_{i0} & \xi & \lambda_2 & & & & & & & & & \\
0 & \omega_{i0} & 0 & \lambda_2 & & & & & & & & \\
v_2 & 0 & \delta_{i1} & \xi & \lambda_2 & & & & & & & \\
 & \mu_2 & 0 & \omega_{i1} & 0 & \lambda_2 & & & & & & \\
 & & 2v_2 & 0 & \delta_{i2} & \xi & \lambda_2 & & & & & \\
 & & & 2\mu_2 & 0 & \omega_{i2} & 0 & \lambda_2 & & & & \\
 & & & & \ddots & & \ddots & \ddots & & & & \\
 & & & & & (c-i)v_2 & 0 & \delta_{i(c-i)} & \xi & \lambda_2 & & \\
 & & & & & & (c-i)\mu_2 & 0 & \omega_{i(c-i)} & 0 & \lambda_2 & \\
 & & & & & & & \ddots & & & \ddots & \\
 & & & & & & & & (c-i)v_2 & 0 & \delta_{iN} & \xi \\
 & & & & & & & & & \omega_{i(c-i)} & 0 & \omega_{iN}
\end{bmatrix},1\le i\le c-1,$$

其中

$$\delta_{ij}=\begin{cases}-\left(\lambda_1+\lambda_2+\xi+iv_1+jv_2\right), & 0\le j<c-i,\\ -\left(\lambda_1+\lambda_2+\xi+iv_1+(c-i)v_2\right), & c-i\le j<N,\\ -\left(\lambda_1+\xi+iv_1+(c-i)v_2\right), & j=N,\end{cases}$$

$$\omega_{ij}=\begin{cases}-\left(\lambda_1+\lambda_2+i\mu_1+j\mu_2\right), & 1\le j<c-i,\\ -\left(\lambda_1+\lambda_2+i\mu_1+(c-i)\mu_2\right), & c-i\le j<N,\\ -\left(\lambda_1+i\mu_1+(c-i)\mu_2\right), & j=N.\end{cases}$$

$$\boldsymbol{A}_c=\begin{bmatrix}
\delta_{i0} & \xi & \lambda_2 & & & & \\
0 & \omega_{10} & 0 & \lambda_2 & & & \\
 & 0 & \delta_{i1} & \xi & \lambda_2 & & \\
 & & 0 & \omega_{11} & 0 & \lambda_2 & \\
 & & & \ddots & \ddots & \ddots & \ddots \\
 & & & & 0 & \delta_{iN} & \xi \\
 & & & & & 0 & \omega_{iN}
\end{bmatrix},$$

其中

$$\delta_{ij}=\begin{cases}-\left(\lambda_1+\lambda_2+\xi+cv_1\right), & 0\le j<N,\\ -\left(\lambda_1+\xi+cv_1\right), & j=N,\end{cases}$$

$$\omega_{ij}=\begin{cases}-\left(\lambda_1+\lambda_2+c\mu_1\right), & 1\le j<N,\\ -\left(\lambda_1+c\mu_1\right), & j=N.\end{cases}$$

3.2. 系统稳态性能分析

对系统的稳态性进行验证，令 $\boldsymbol{H}=\boldsymbol{B}_c+\boldsymbol{A}_c+\boldsymbol{C}_1$ 是有限不可约生成元，$\boldsymbol{\pi}$ 为 $\boldsymbol{H}$ 的平稳概率向量，其中由文献[18]可知系统存在稳态分布的条件为 $\boldsymbol{\pi}\boldsymbol{B}_c\boldsymbol{e}>\boldsymbol{\pi}\boldsymbol{C}_1\boldsymbol{e}$。将 $\boldsymbol{\pi}\boldsymbol{H}=0,\boldsymbol{\pi}\boldsymbol{e}=1$ 代入 $\boldsymbol{\pi}\boldsymbol{B}_c\boldsymbol{e}>\boldsymbol{\pi}\boldsymbol{C}_1\boldsymbol{e}$ 中，可得到当 $\rho=\lambda_1/c\mu_1<1$ 时，MarKov 过程 $\left\{\left(N_1(t),N_2(t),S(t)\right),t\geq 0\right\}$ 是正常返的，即稳态分布存在。系统稳态分布可以表示为：

$$\pi_{ijs}=\lim_{t\to\infty}P\left\{N_1(t)=i,N_2(t)=j,S(t)=s\right\},\quad (i,j,s)\in\Omega$$

记稳态概率向量为

$$\boldsymbol{\pi}_0=\left(\pi_{000},\pi_{010},\pi_{011},\pi_{020},\pi_{021},\cdots,\pi_{0N0},\pi_{0N1}\right)$$

$$\boldsymbol{\pi}_k=\left(\pi_{k00},\pi_{k01},\pi_{k10},\pi_{k11},\pi_{k20},\pi_{k21},\cdots,\pi_{kN0},\pi_{kN1}\right)$$

由拟生灭过程与矩阵几何解方法可知，求解系统稳态分布需要先求出矩阵方程 $\boldsymbol{R}^2\boldsymbol{B}_c+\boldsymbol{R}\boldsymbol{A}_c+\boldsymbol{C}_1=0$ 的最小非负解 $\boldsymbol{R}$，且谱半径 $SP(\boldsymbol{R})<1$，同时 $2N(c+1)+2c+1$ 维矩阵

$$\boldsymbol{B}[\boldsymbol{R}]=\begin{bmatrix}\boldsymbol{A}_0 & \boldsymbol{C}_0 & & & & & & \\ \boldsymbol{B}_1 & \boldsymbol{A}_1 & \boldsymbol{C}_1 & & & & & \\ & \boldsymbol{B}_2 & \boldsymbol{A}_2 & \boldsymbol{C}_1 & & & & \\ & & \boldsymbol{B}_3 & \boldsymbol{A}_3 & \boldsymbol{C}_1 & & & \\ & & & \ddots & \ddots & \ddots & & \\ & & & & \boldsymbol{B}_{c-1} & \boldsymbol{A}_{c-1} & \boldsymbol{C}_1 & \\ & & & & & \boldsymbol{B}_c & \boldsymbol{R}\boldsymbol{B}_c+\boldsymbol{A}_c & \end{bmatrix}$$

有左零向量，且稳态分布满足如下方程组

$$\boldsymbol{\pi}_0\boldsymbol{A}_0+\boldsymbol{\pi}_1\boldsymbol{B}_1=0, \tag{1}$$

$$\boldsymbol{\pi}_0\boldsymbol{C}_0+\boldsymbol{\pi}_1\boldsymbol{A}_1+\boldsymbol{\pi}_2\boldsymbol{B}_2=0, \tag{2}$$

$$\boldsymbol{\pi}_k\boldsymbol{C}_1+\boldsymbol{\pi}_{k+1}\boldsymbol{A}_{k+1}+\boldsymbol{\pi}_{k+2}\boldsymbol{A}_{k+2}=0,\ 1\leq k\leq c-2, \tag{3}$$

$$\boldsymbol{\pi}_{c-1}\boldsymbol{C}_1+\boldsymbol{\pi}_c\left(\boldsymbol{R}\boldsymbol{B}_c+\boldsymbol{A}_c\right)=0, \tag{4}$$

$$\sum_{n=0}^{\infty}\boldsymbol{\pi}_n\boldsymbol{e}=1, \tag{5}$$

$$\boldsymbol{\pi}_n=\boldsymbol{\pi}_c\boldsymbol{R}^{n-c},\ n\geq c, \tag{6}$$

其中 $\boldsymbol{e}$ 为全 1 列向量。

然而高阶矩阵方程 $\boldsymbol{R}^2\boldsymbol{B}_c+\boldsymbol{R}\boldsymbol{A}_c+\boldsymbol{C}_1=0$ 很难直接求出率阵 $\boldsymbol{R}$ 的表达式，下面采用 Neuts [18]给出的迭代算法来求解率阵 $\boldsymbol{R}$。为保证算法的收敛性，给出算法的精度 ε，当 $\left\|\boldsymbol{R}_{n+1}-\boldsymbol{R}_n\right\|<\varepsilon$ 时，得到 $\boldsymbol{R}$ 的近似解，具体迭代步骤如下

Step 1：对系统的参数 $\lambda_1,\lambda_2,\mu_1,\mu_2,\mu_{v1},\mu_{v2},\xi,c,N$，定义初始值，设置精度 $\varepsilon=10^{-10}$，初始化率阵 $\boldsymbol{R}=0$

Step 2：输入 $\boldsymbol{B}_c,\boldsymbol{A}_c,\boldsymbol{C}_1$

Step 3：定义 $\boldsymbol{R}_n=\boldsymbol{R}$

$\boldsymbol{R}=-\left(\boldsymbol{R}^2\boldsymbol{B}_c+\boldsymbol{C}_1\right)\boldsymbol{A}_c^{-1}$

$\boldsymbol{R}_{n+1}=\boldsymbol{R}$

Step 4：如果 $\left\|\boldsymbol{R}_{n+1}-\boldsymbol{R}_n\right\|>\varepsilon$ 则返回 Step 3，否则进入 Step 5

Step 5： $\boldsymbol{R}=\boldsymbol{R}_{n+1}$

通过上述算法求解出率阵 $\boldsymbol{R}$ 后，由(1)~(6)可得到如下递推方程

$$\boldsymbol{\pi}_0=\boldsymbol{\pi}_1\boldsymbol{B}_1\left(-\boldsymbol{A}_0\right)^{-1}=\boldsymbol{\pi}_1\boldsymbol{\beta}_1, \tag{7}$$

$$\boldsymbol{\pi}_1=\boldsymbol{\pi}_2\boldsymbol{B}_2\left(-\left(\boldsymbol{\beta}_1\boldsymbol{C}_0+\boldsymbol{A}_1\right)\right)^{-1}=\boldsymbol{\pi}_2\boldsymbol{\beta}_2, \tag{8}$$

$$\boldsymbol{\pi}_k=\boldsymbol{\pi}_{k+1}\boldsymbol{B}_{k+1}\left(-\left(\boldsymbol{\beta}_k\boldsymbol{C}_1+\boldsymbol{A}_k\right)\right)^{-1}=\boldsymbol{\pi}_{k+1}\boldsymbol{\beta}_{k+1},\quad 2\le k\le c-1, \tag{9}$$

$$\boldsymbol{\pi}_c\boldsymbol{\beta}_c\boldsymbol{C}_1+\boldsymbol{\pi}_c\left(\boldsymbol{R}\boldsymbol{B}_c+\boldsymbol{A}_c\right)=0, \tag{10}$$

$$\sum_{n=0}^{c-1}\boldsymbol{\pi}_n e+\boldsymbol{\pi}_c\left(\boldsymbol{I}-\boldsymbol{R}\right)^{-1}\boldsymbol{e}=1. \tag{11}$$

由(7)~(11)可以递推地求出稳态概率向量。

4. 系统稳态性能指标

基于系统的稳态分布，得到在抢占优先权和同步半休眠模式下 VM 调度策略的稳态性能指标：

1) 第一类顾客的平均队长

$$E\left(L_1\right)=\sum_{i=0}^{\infty}iP\left(L_1=i\right)=\sum_{i=1}^{\infty}\sum_{s=0}^{1}\sum_{j=0}^{N}i\pi_{ijs}$$

$$=\sum_{i=1}^{c-1}i\boldsymbol{\pi}_i+c\boldsymbol{\pi}_c\left(\boldsymbol{I}-\boldsymbol{R}\right)^{-1}+\boldsymbol{\pi}_c\boldsymbol{R}\left(\boldsymbol{I}-\boldsymbol{R}\right)^{-2}.$$

2) 第二类顾客的平均队长

$$E\left(L_2\right)=\sum_{j=0}^{N}jP\left(L_2=j\right)=\sum_{j=1}^{N}j\sum_{s=0}^{1}\sum_{i=0}^{\infty}\pi_{ijs}.$$

3) 服务台的利用率

$$P_u=\frac{\min\left\{E\left(L_1\right)+E\left(L_2\right),c\right\}}{c}.$$

4) 内存溢出率

$$P_l=\sum_{i=0}^{\infty}\sum_{s=0}^{1}\sum_{j=N}^{\infty}\pi_{ijs}=\sum_{i=0}^{\infty}\sum_{s=0}^{1}\pi_{iNs}.$$

5) 处于工作状态和半休眠状态的 VM 的总能耗

$$W=cW_1\sum_{i=0}^{\infty}\sum_{j=1}^{N}\pi_{ij1}+cW_2\sum_{i=0}^{\infty}\sum_{j=0}^{N}\pi_{ij0}+W_3\left[E\left(L_1\right)+E\left(L_2\right)\right].$$

其中一台 VM 在工作状态时的能耗为 W_1，在半休眠状态时的能耗为 W_2，两类顾客在缓冲器中排队等待的能耗为 W_3。

5. 数值分析

在实际背景中，系统各稳态性能指标常因参数的变化而有所不同。本节通过数值模拟研究各参数对系统稳态性能指标的影响，令 $\lambda_1=8,\lambda_2=10,\mu_2=3,\nu_1=1,\nu_2=0.5,\xi=3,N=50$ 。

见图 1 研究了第二类顾客的平均队长 $E\left(L_2\right)$ 与 μ_1 和 c 的变化关系。当 $c=4$ 时且 $3\le\mu_1\le4$ 时，系统中为第一类顾客服务的 VM 数较多，第二类顾客接受服务的机会较小，$E\left(L_2\right)$ 随 μ_1 的增大而增大。当 $c=4$

时且$4<\mu_1\le 8$时，随着第一类顾客服务率的增大，第二类顾客接受服务的机会增大，$E(L_2)$随μ_1的增大而减小。当$c=5$或$c=6$时，$E(L_2)$随μ_1的增大而减小。当μ_1一定时，VM 数的增大使得第二类顾客接受服务的机会增大，$E(L_2)$随c的增大而减小。

见图 2 研究了服务台利用率P_u与μ_1和c的变化关系。当$c=4$时，c台 VM 均处于为顾客服务的状态，服务台的利用率为 1。当$c=5$或$c=6$时，随着第一类顾客服务率的增大，顾客在系统中的平均逗留时间越小，P_u随μ_1的增大而逐渐减小。当μ_1一定时，VM 数的增多使得顾客接受服务的机会增大，P_u随c的增大而减小。系统应适当调整 VM 数和服务率来实现队长、服务台利用率的优化。

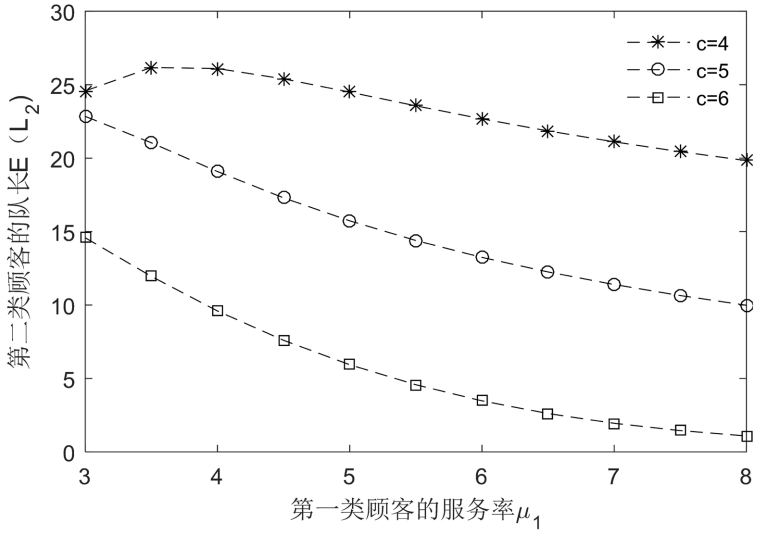

Figure 1. The change of $E(L_2)$ with respect to μ_1 and c

图 1. $E(L_2)$随μ_1和c的变化关系

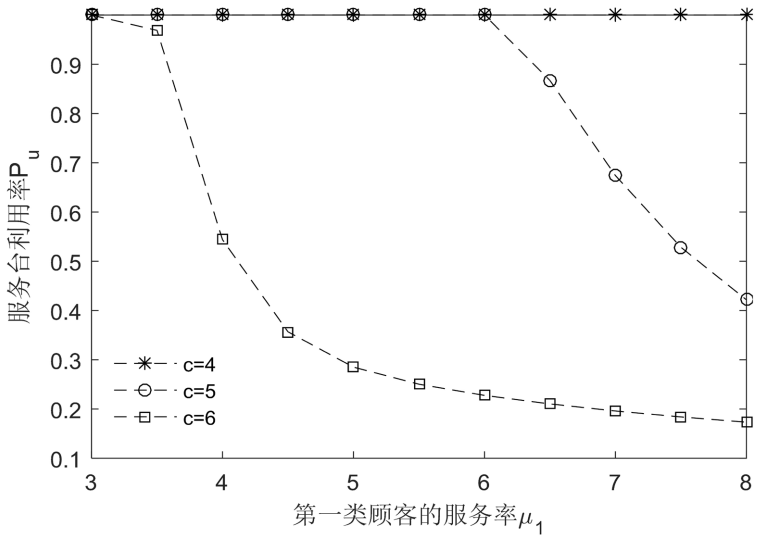

Figure 2. The change of P_u with respect to μ_1 and c

图 2. P_u随μ_1和c的变化关系

见表 1 研究了在$c=4$的条件下，内存溢出率P_l与N和μ_1的变化关系。当N一定且μ_1较小时，第二

类顾客接受服务的机会较少，发生内存溢出的概率增大。当 N 一定且 μ_1 较大时，第二类顾客接受服务的机会增大，因此 P_l 随 μ_1 的增大先增大后减小。当 μ_1 一定时，N 越大系统可以容纳的第二类顾客数越多，发生内存溢出的概率减小，因此 P_l 减小。系统可以通过适当调整系统容量和服务率来减少内存溢出。

Table 1. The change of P_l with respect to N and μ_1
表 1. P_l 随 N 和 μ_1 的变化关系

N	μ_1	P_l	N	μ_1	P_l
50	3	0.491 48	53	3	0.026 82
50	3.5	0.523 39	53	3.5	0.037 75
50	4	0.521 92	53	4	0.047 17
50	4.5	0.507 94	53	4.5	0.055 24
50	5	0.489 87	53	5	0.062 11
50	5.5	0.471 23	53	5.5	0.067 96
50	6	0.453 45	53	6	0.072 93
51	3	0.145 57	54	3	0.012 49
51	3.5	0.178 59	54	3.5	0.018 53
51	4	0.199 23	54	4	0.024 19
51	4.5	0.212 19	54	4.5	0.029 42
51	5	0.220 24	54	5	0.034 17
51	5.5	0.225 06	54	5.5	0.038 46
51	6	0.227 75	54	6	0.042 30
52	3	0.059 90	55	3	0.005 93
52	3.5	0.079 33	55	3.5	0.009 23
52	4	0.094 23	55	4	0.012 55
52	4.5	0.105 74	55	4.5	0.015 81
52	5	0.114 67	55	5	0.018 93
52	5.5	0.121 63	55	5.5	0.021 88
52	6	0.127 07	55	6	0.024 64

见图 3 研究了内存溢出率 P_l 与 μ_1 和 c 的变化关系。当 $c=4$ 时且 $3\le\mu_1\le3.5$ 时，系统中为第二类顾客服务的 VM 数较少，系统中等待的第二类顾客数增多，发生内存溢出的概率增大，P_l 随 μ_1 的增大而增大。当 $c=4$ 时且 $3.5<\mu_1\le8$ 时，第二类顾客接受服务的机会增大，发生内存溢出的概率减小，P_l 随 μ_1 的增大而减小。当 $c=5$ 或 $c=6$ 时，P_l 随 μ_1 的增大而减小。当 μ_1 一定时，数数的增大使得第二类顾客接受服务的机会增大，P_l 随 c 的增大而减小。

见图 4 研究了在 $\mu_1=6$ 系统总能耗 W 与 μ_2 和 c 的变化关系。当 c 一定时，μ_2 增大使系统中第二类顾客在缓冲器中排队等待的能耗减少，因此 W 随 μ_2 的增大而减小。当 μ_2 一定时，VM 数越多，系统总能耗增加，因此 W 随 c 的增大而增大。

见表 2 研究了在 c 台 VM 同步半休眠策略下，半休眠参数 ξ 对系统总能耗 W 的影响，随着半休眠参数 ξ 的增大，VM 处于半休眠状态的时间越短，系统总能耗 W 随 ξ 的增大而增大。当 ξ 一定时，系统中处于正常工作状态或者半休眠状态的 VM 数越多耗能就越大，因此系统总能耗 W 随 VM 数的增大而增大。系统应适当调整 VM 数和工作休假参数来实现系统总能耗的优化。

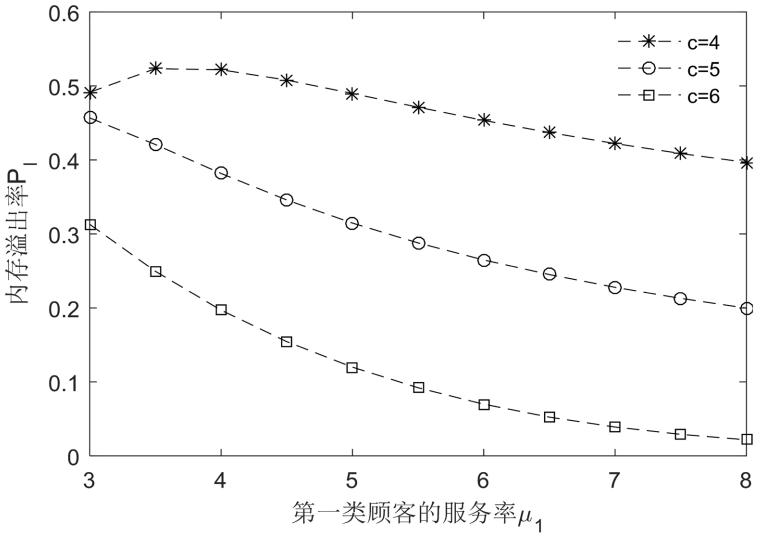

Figure 3. The change of P_l with respect to μ_1 and c
图 3. P_l 随 μ_1 和 c 的变化关系

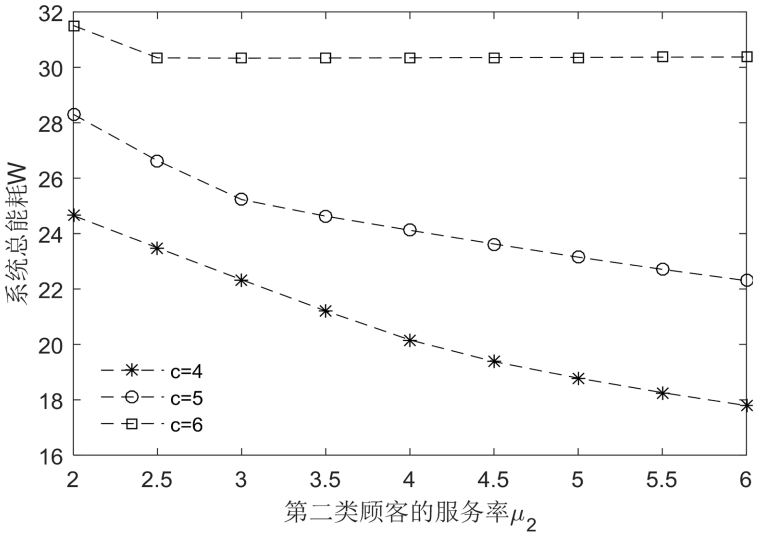

Figure 4. The change of W with respect to μ_2 and c
图 4. W 随 μ_2 和 c 的变化关系

Table 2. The impact of system parameters ξ and c on W
表 2. 系统参数 ξ 和 c 对 W 的影响

ξ	$c=4$	$c=5$	$c=6$
0.5	21.866 53	24.179 91	27.298 00
1.0	22.347 58	24.670 96	27.839 05
1.5	22.644 83	24.978 21	28.196 30
2.0	22.841 25	25.184 63	28.452 72
2.5	22.976 73	25.330 11	28.648 20
3.0	23.071 82	25.435 20	28.803 29

续表

3.5	23.138 30	25.511 68	28.929 78
4.0	23.183 57	25.566 95	29.035 04

6. 系统收益优化

根据云数据中心服务一位顾客就会得到一份收益，设 R_1 和 R_2 分别表示服务完一位高优先级顾客和一位低优先级顾客所获得的收益，C_1 和 C_2 分别表示高优先级顾客和低优先级顾客在系统内单位逗留时间所花费的成本，C_3 表示系统单位能耗所带来的损失，C_4 表示因内存空间有限所造成的内存溢出的损失。从社会的角度考虑整个系统，定义系统收益 U_s 为系统中第一类顾客和第二类顾客完成服务后所获得到的总收益再减去因系统能耗和内存空间有限所造成的内存溢出的损失，具体表达式如下：

$$U_s = \lambda_1\left[R_1 - C_1\frac{E(L_1)}{\lambda_1}\right] + \lambda_2\left[R_2 - C_2\frac{E(L_2)}{\lambda_2}\right] - C_3W - C_4P_l$$

见图 5 研究了在 $c=4, R_1=25, R_2=18, C_1=2, C_2=1.2, C_3=3, C_4=4$ 的条件下，系统收益 U_s 与 μ_1 和 c 之间的关系。当 $c=4$ 且 $3\le\mu_1\le4$ 时，系统中第二类顾客的平均队长增加，发生内存溢出的概率增大且因内存溢出所造成的损失增加，因此 U_s 随 μ_1 的增大而减小。当 $c=4$ 且 $4<\mu_1\le6$ 时，第二类顾客因被抢占 VM 而终止服务的概率减小，系统因服务完更多地顾客而获得了更多地收益，因此 U_s 随 μ_1 的增大而增大。当 $c=5$ 或 $c=6$ 时，U_s 随 μ_1 的增大而增大。当 μ_1 $(3\le\mu_1\le5)$ 一定时，U_s 随 c 的增大而减小。当 μ_1 $(5<\mu_1\le6)$ 一定时，VM 数为 5 时 U_s 最大。

见图 6 研究了在 $c=4, R_1=25, R_2=18, C_1=2, C_2=1.2, C_3=3, C_4=4$ 的条件下，系统收益 U_s 与 μ_1 和 N 之间的关系。当 μ_1 一定时，第二类顾客的等待空间 N 越大，发生内存溢出的概率就越小，因此 U_s 随 N 的增大而增大。当 N 一定时，发生内存溢出的概率先增大减小，所造成的损失相应的先增大后减小。

见表 3 研究了在 $\mu_1=6$，系统收益 U_s 与 N 和 μ_2 之间的关系，当 N 一定，随着 μ_2 的增大，第二类顾客在系统中的平均逗留时间减少，系统服务的顾客数增多，因此 U_s 随 μ_2 的增大而增大。当 μ_2 $(\mu_2=2)$ 一定时，第一类顾客具有抢占优先权且第二类顾客的服务率较小，使得第二类顾客接受服务的机会较小，

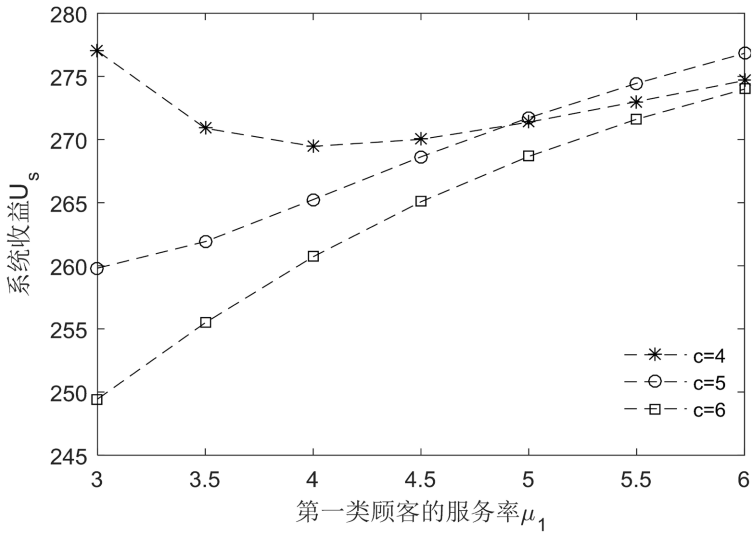

Figure 5. The change of U_s with respect to μ_1 and c

图 5. U_s 随 μ_1 和 c 的变化关系

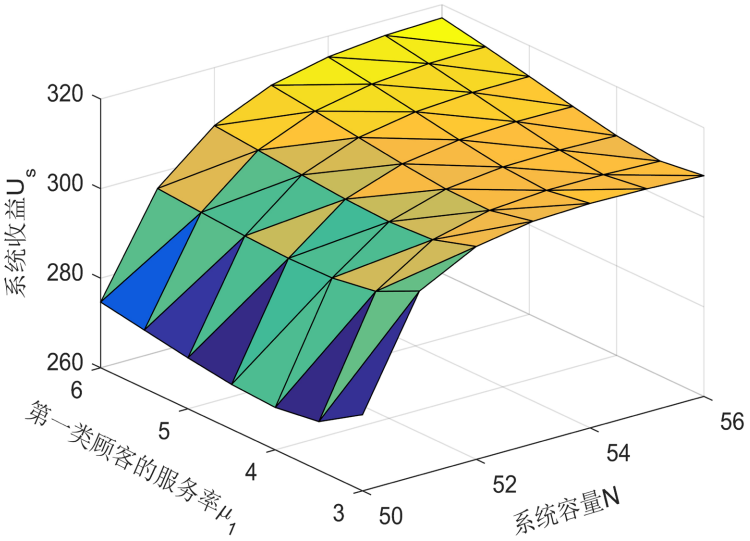

Figure 6. The change of U_s with respect to μ_1 and N
图 6. U_s 随 μ_1 和 N 的变化关系

顾客在系统内因逗留时间过长导致系统成本增加，因此 U_s 随 N 的增大而减小。当 μ_2 $(2<\mu_2\le 3.5)$一定时，服务率较大使得第二类顾客在系统中平均逗留时间较小，且内存空间 N 的增大使得第二类顾客有更多的机会接受服务，且因内存溢出所造成的损失减少，U_s 随 N 的增大而增大。因此在第一类顾客的服务率一定的情况下，系统应当适当扩大内存空间和提高第二类顾客服务率，使得系统收益增多。

Table 3. The impact of system parameters N and μ_2 on U_s
表 3. 系统参数 N 和 μ_2 对 U_s 的影响

N	μ_2	U_s	N	μ_2	U_s
55	2	313.162 37	58	2	312.174 21
55	2.5	320.920 58	58	2.5	322.913 49
55	3	325.355 25	58	3	330.509 51
55	3.5	325.378 49	58	3.5	331.846 26
56	2	313.122 86	59	2	311.454 67
56	2.5	322.128 95	59	2.5	323.522 56
56	3	327.616 32	59	3	331.368 27
56	3.5	327.805 80	59	3.5	333.482 14
57	2	312.764 65	60	2	310.663 27
57	2.5	322.726 77	60	2.5	323.543 89
57	3	329.291 74	60	3	331.944 91
57	3.5	329.965 18	60	3.5	334.901 36

7. 结论

本文基于所提出的 VM 同步半休眠策略，研究了一种基于抢占优先权且服务台同步工作休假的 M/M/c 排队模型。通过对系统进行描述，建立了三维 Markov 过程，得到其拟生灭过程及无穷小生成元矩

阵，采用拟生灭过程理论求解出系统的稳态分布，得出系统中两类顾客的平均队长、内存溢出率、系统总能耗等稳态性能指标，并通过数值模拟讨论系统稳态性能指标随参数变化的关系，研究 c 台 VM 同步半休眠策略对系统总能耗的影响。最后，建立系统收益函数，研究参数对系统收益函数的作用规律。从而在保证两类顾客服务质量的前提下，通过适当调整 VM 数、内存空间和服务率增加系统收益。

参考文献

[1] White, H. and Christie, L. (1958) Queueing with Preemptive Priorities with Breakdown. *Operations Research*, **6**, 79-95. https://doi.org/10.1287/opre.6.1.79

[2] Rao, S. (1967) Queueing with Balking and Reneing in M/G/1 Systems. *Metrika*, **12**, 173-188. https://doi.org/10.1007/BF02613493

[3] Gail, H., Hantler, S. and Tayler, B. (1992) On a Preemptive Markovian Queue with Multiple Servers and Two Priority Classes. *Mathematics of Operations Research*, **17**, 365-391. https://doi.org/10.1287/moor.17.2.365

[4] 徐秀丽, 高红, 田乃硕. 对带启动时间和可变服务率的 M/M/1 休假排队的分析[J]. 应用数学学报, 2008, 31(4): 692-701.

[5] 徐秀丽. (e, d)型休假 M/M/c 排队的稳态理论及应用[D]: [博士学位论文]. 秦皇岛: 燕山大学, 2006.

[6] 郭闪闪. 基于抢占优先的部分服务台工作休假排队系统的研究[D]: [硕士学位论文]. 秦皇岛: 燕山大学, 2020.

[7] Si, Q.N., Ma, Z.Y., Liu, F.J., *et al*. (2021) Performance Analysis of P2P Network with Dynamic Changes of Servers Based on M/M/c Queuing Model. *Wireless Networks*, **27**, 3287-3297. https://doi.org/10.1007/s11276-021-02659-2

[8] Sahoo, C.N. and Goswami, V. (2017) Cost and Energy Optimisation of Cloud Data Centres through Dual VM Modes-Activation and Passivation. *International Journal of Communication Networks and Distributed Systems*, **18**, 371-389. https://doi.org/10.1504/IJCNDS.2017.10004670

[9] 李吉良, 秦兵, 李文江, 等. 融合唤醒阈值与半休眠模式的云虚拟机调度策略[J]. 燕山大学学报, 2020, 44(4): 370-378.

[10] 金顺福, 郄修尘, 武海星, 等. 基于新型休眠模式的云虚拟机分簇调度策略及性能优化[J]. 吉林大学学报(工学版), 2020, 50(1): 237-246.

[11] Guo, M., Guan, Q.S., Chen, W.Q., *et al*. (2022) Delay-Optimal Scheduling of VMs in a Queueing Cloud Computing System with Heterogeneous Workloads. *IEEE Transactions on Services Computing*, **15**, 110-123. https://doi.org/10.1109/TSC.2019.2920954

[12] Ma, Z.Y., Guo, S.S. and Wang, R. (2023) The Virtual Machines Scheduling Strategy Based on M/M/c Queueing Model with Vacation. *Future Generation Computer Systems*, **138**, 43-51. https://doi.org/10.1016/j.future.2022.08.001

[13] Jin, S.F., Hao, S.S., Qie, X.C., *et al*. (2019) A Virtual Machine Scheduling Strategy with a Speed Switch and a Multi-Sleep Mode in Cloud Data Centers. *Journal of Systems Science and Systems Engineering*, **28**, 194-210. https://doi.org/10.1007/s11518-018-5401-9

[14] Jin, S.F., Qie, X.C., Zhao, W.J., *et al*. (2020) A Clustered Virtual Machine Allocation Strategy Based on a Sleep-Mode with Wake-Up Threshold in a Cloud Environment. *Annals of Operations Research*, **293**, 193-212. https://doi.org/10.1007/s10479-019-03339-3

[15] Li, W. and Jin, S.F. (2021) Nash Equilibrium and Social Optimization in Cloud Service Systems with Diverse Users. *Cluster Computing*, **24**, 2039-2050. https://doi.org/10.1007/s10586-021-03242-2

[16] Li, W. and Jin, S.F. (2021) Performance Evaluation and Optimization of a Task Offloading Strategy on the Mobile Edge Computing with Edge Heterogeneity. *Journal of Supercomputing*, **77**, 12486-12507. https://doi.org/10.1007/s11227-021-03781-w

[17] Cui, Y., Zhang, Y., Li, X., *et al*. (2023) A Dynamic Energy Conservation Scheme with Dual-Rate Adjustment and Semi-Sleep Mode in Cloud System. *The Journal of Supercomputing*, **79**, 2451-2487. https://doi.org/10.1007/s11227-022-04715-w

[18] Neuts, M. (1981) Matrix Geometric Solution on Stochastic Models. Johns Hokpins University Press, Baltimore, 293-309.