

DOI: 10.19783/j.cnki.pspc.210466

# 基于改进蚁群算法的电力云数据中心任务调度策略研究

孙湛冬<sup>1</sup>, 焦娇<sup>1</sup>, 李伟<sup>1</sup>, 李志鹏<sup>1</sup>, 李鹏恩<sup>2</sup>

(1. 北京电力经济技术研究院有限公司, 北京 100055; 2. 北京恒华伟业科技股份有限公司, 北京 100011)

**摘要:** 针对现有云环境下电力数据中心任务调度的高能耗、低效率等问题, 在电力云体系结构的基础上, 提出了一种基于随机 Petri 网的云数据中心任务调度模型。通过综合考虑时间约束、负载、能耗约束对蚁群算法进行改进, 并通过改进算法对模型进行求解。通过实验对运行时间、能耗、平均等待时间、系统负载等几个方面进行了比较分析, 验证了该方法的优越性。结果表明, 改进蚁群算法在保证性能的前提下, 可以有效降低数据中心能耗, 为电力数据中心任务调度策略的发展提供参考和借鉴。

**关键词:** 云环境; 任务调度; Petri 网; 蚁群优化算法; 电力数据; 运行能耗

## A task scheduling strategy for a power cloud data center based on an improved ant colony algorithm

SUN Zhandong<sup>1</sup>, JIAO Jiao<sup>1</sup>, LI Wei<sup>1</sup>, LI Zhipeng<sup>1</sup>, LI Peng'en<sup>2</sup>

(1. Beijing Electric Power Economic Research Institute Co., Ltd., Beijing 100055, China;  
2. Beijing Forever Technology Co., Ltd., Beijing 100011, China)

**Abstract:** There are problems of high energy consumption and low efficiency of power data center task scheduling in the existing cloud environment. Thus, based on power cloud architecture, this paper proposes a cloud data center task scheduling based on the stochastic Petri net model, and considers the time, load, and energy consumption constraints to improve the ant colony algorithm to solve the problem presented by the model. The advantages of this method are verified by comparing and analyzing the running time, energy consumption, average waiting time and system load. The results show that the improved ant colony algorithm can effectively reduce the energy consumption of the data center and guarantee performance. It provides a reference for the development of a task scheduling strategy in a power data center.

This work is supported by the Science and Technology Project of State Grid Corporation of China (No. 520234190006).

**Key words:** cloud environment; task scheduling; Petri network; ant colony optimization algorithm; power data; operation energy consumption

## 0 引言

信息平台是支撑统一坚强智能电网建设的公共平台和重要手段, 从而为智能电网提供了更高效、灵活、安全可靠的信息基础架构<sup>[1]</sup>。随着电力信息技术的不断发展, 数据量的不断增加和云计算的出现, 大数据处理成为可能。云计算根据需要获取计算能力和存储空间等服务, 广泛应用于智能电网领域。电力调度是整个电网的大脑, 需要高效、安全的大数据计算能力支撑<sup>[2]</sup>。因此, 研究云环境下的

电力数据中心任务调度策略具有重要的实际意义。

目前, 国内外学者对电力云数据中心调度策略进行了大量的研究, 取得了部分研究成果。文献[3]提出了一种云任务调度优化策略, 该策略综合考虑了用户的最短等待时间、资源负载分配和经济原则, 建立了多目标数学模型, 通过离散的人工蜂群算法进行求解。结果表明, 该方法可以较好地提高虚拟资源上云任务调度系统的性能。文献[4]提出了一种改进的蚁群算法, 用于电力云任务调度策略。结果表明, 改进算法在保证数据集处理的基础上, 可以有效降低执行时间和均衡负载。文献[5]提出了一种改进的遗传算法, 用于电力云任务调度策略。该策

基金项目: 国家电网公司科技项目资助(520234190006)

略对最大任务和平均任务完成时间进行考虑,使系统可靠性和用户满意度得到加强。文献[6]提出了一种改进的蚁群算法,用于电力云任务调度策略,以解决虚拟机的任务调度问题。通过对基本蚁群算法改进实现任务调度时间和负载均衡,从而最小化总体完成时间。结果表明,该算法可以减少任务调度时间,实现云节点负载均衡。文献[7]提出一种基本蚁群算法,用于电力云任务调度策略,结果表明,该算法可以较为灵活地进行任务处理,同时减少任务完成时间。文献[8]提出一种新的聚类调度算法,结果表明,该算法考虑了系统特性,可以实现高效的调度。但是,上述方法没有考虑能耗问题,因此需要进一步提高解的质量和适应性。

因此本文提出了一种基于改进蚁群算法的电力云数据中心调度策略,使用随机 Petri 网对云数据中心任务调度进行建模,通过改进蚁群优化(Ant Colony Optimization, ACO)算法求解该模型。仿真验证了该方法的优越性。

### 1 电力云架构

云计算广泛应用于智能电网领域,在电网建设、智能分析、存储等方面作用巨大[9]。云计算技术在电力数据中心的应用可以提高服务器资源利用效率。电力云不是单一的服务,而是多种服务的集合,图 1 为电力云架构。它主要由五层组成:基础结构层、虚拟化层、系统管理层、应用接口层和服务访问层[10]。



图 1 系统架构

Fig. 1 System architecture

## 2 调度策略

### 2.1 建模

文中采用随机 Petri 网对云数据中心的任务调度进行建模[11]。首先对任务进行分解,通过调度将子任务分配给计算资源。

对于一组虚拟机计算资源  $VM = \{VM_1, VM_2, \dots, VM_n\}$ ,  $VM_j$  为第  $j$  个虚拟机资源,要执行的一组任务组  $T = \{T_1, T_2, \dots, T_m\}$ , 资源使用可用  $m \times n$  矩阵  $R_{m \times n}$  表示, 矩阵如式(1)所示[12]。

$$R_{m \times n} = \begin{pmatrix} R_{11} & R_{12} & \dots & R_{1n} \\ R_{21} & R_{22} & \dots & R_{2n} \\ \vdots & \vdots & & \vdots \\ R_{m1} & R_{m2} & \dots & R_{mn} \end{pmatrix} \quad (1)$$

式中,  $R_{ij}$  表示任务  $T_i$  在虚拟机  $VM_j$  上运行。

连续时间随机 Petri 网可以用六元组来表示,如式(2)所示[13]。

$$SPN = (P, T, F, W, M_0, \lambda) \quad (2)$$

式中:  $P$  为库所集;  $T$  为变迁集合;  $F$  为网的流关系;  $W$  为有向边集合上的权函数;  $M_0$  为系统的初始识别;  $\lambda$  为变迁的平均实施率的集合  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_m)$ , 为变迁  $i$  的平均服务时间。图 2 为云任务调度系统的随机 Petri 网模型。

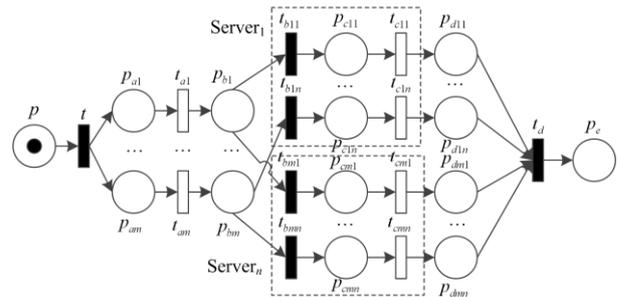


图 2 系统随机 Petri 网模型

Fig. 2 Systematic stochastic Petri net model

任务分解是将所负责任务分解成独立的  $m$  个子任务,对应于 Map 任务[14]。由于该过程的处理时间短,因此,随机 Petri 网模型中,瞬时变迁用  $t$  表示。任务分解后,将执行队列输入操作,该过程由变迁  $t_a$  表示。主程序将子任务平均分配给 Worker 机器。分配包括库所  $p_b$  和瞬时变迁  $t_b$ , 可以采用不同调度方案。分配完任务后, Worker 机器对任务进行处理,用变迁  $t_c$  表示,由调度方案确定任务处理时间。生成中间结果集  $p_d$ 。最后,合并完成每个任务后获得的中间结果集,以获取最终结果集  $p_e$ ,并用变

迁 $t_d$ 表示。

模型只对时间参数的变迁进行考虑, 时间变迁 $t_{a1}, \dots, t_{am}$ , 构成一个并发结构, 假设变迁 $t_{ai}$ 的平均实施延迟为 $1/\lambda_{a1}, \dots, 1/\lambda_{am}$ 。根据并发结构和时间性能等价公式, 等效时间变迁 $t_a$ 的平均延迟时间为 $1/\lambda_a$ 。计算如式(3)所示<sup>[15]</sup>。

$$\frac{1}{\lambda_a} = \sum_{i=1}^m \frac{1}{\lambda_{ai}} - \sum_{i=1}^{m-1} \sum_{j=i+1}^m \frac{1}{\lambda_{ai} + \lambda_{aj}} + \dots + (-1)^{n-1} \frac{1}{\sum_{i=1}^m \lambda_{ai}} \quad (3)$$

假设变迁的平均实施延迟为 $1/\lambda_{c11}, \dots, 1/\lambda_{c1n}, \dots, 1/\lambda_{cm1}, \dots, 1/\lambda_{cmn}$ , 执行概率为 $\alpha_{ij}, 1 \leq i \leq m, 1 \leq j \leq n$ , 且 $\sum_{i=1}^m \sum_{j=1}^n \alpha_{ij} = 1$ 。根据选择结构和时间性能的等价公式, 等效时间变迁 $t_c$ 的平均延迟为 $1/\lambda_c$ , 计算结果如式(4)所示<sup>[16]</sup>。

$$\frac{1}{\lambda_c} = \sum_{i=1}^m \sum_{j=1}^n \frac{\alpha_{ij}}{\lambda_{ij}} \quad (4)$$

通过上述方法, 可以获得整个系统的性能等价时间。

## 2.2 蚁群算法

区域栅格化形成一个网络, 并给出每个栅格的分类和成本值。蚁群算法搜索流程如图3所示。

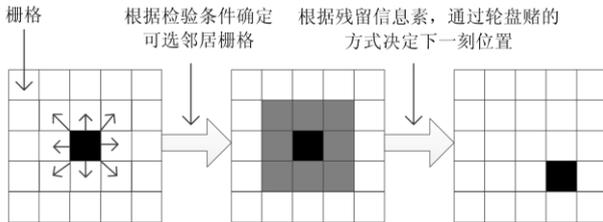


图3 蚁群算法搜索流程

Fig. 3 Ant colony algorithm search process

蚁群算法的搜索过程主要使用两种类型的规则, 确定下一个位置移动规则和算法的核心信息素更新规则<sup>[17]</sup>。在信息素的引导下, 蚁群解向最佳方向发展。

如图3所示, 可以假设蚂蚁的位置(黑色)可以移至下一个位置, 例如B的灰色部分。蚂蚁的移动方向与移动规则有关, 该运动规则由路径的信息素浓度和相邻栅格之间的距离确定。

假设蚂蚁的下一个位置是 $i$ , 式(5)表示从 $o$ 到 $i$ 的概率<sup>[18]</sup>。

$$p_{o \rightarrow i} = c_{o \rightarrow i}^\alpha \tau_{o \rightarrow i}^\beta \quad (5)$$

归一化处理如式(6)所示<sup>[19]</sup>。

$$p'_{o \rightarrow i} = \frac{c_{o \rightarrow i}^\alpha \tau_{o \rightarrow i}^\beta}{\sum_{i=1}^n c_{o \rightarrow i}^\alpha \tau_{o \rightarrow i}^\beta} \quad (6)$$

式中:  $c_{o \rightarrow i}$ 为从 $o$ 到 $i$ 栅格之间距离的倒数;  $\tau_{o \rightarrow i}$ 为 $i$ 栅格信息素浓度;  $\alpha$ 、 $\beta$ 分别为控制强度;  $n$ 为可选的相邻栅格数。

如式(6)所示, 每个栅格被选择的概率 $r$ 是由计算机随机生成的, 介于0和1之间。使用轮盘赌方法, 即 $\sum_{i=1}^n p'_{o \rightarrow i} \geq r$ 时, 相邻栅格 $i$ 为下一个位置。一

旦找到可行的路径, 就需要更新网格信息素浓度。同时, 引入信息素挥发性因子以防止信息素浓度过高, 从而防止路径早熟。栅格信息素更新如式(7)所示<sup>[20]</sup>。

$$\tau'_i = (1 - \sigma)(\tau_i + \Delta\tau) \quad (7)$$

式中:  $\sigma$ 为信息素的挥发系数,  $\sigma \in (0, 1)$ ;  $\tau_i$ 为信息素更新前的浓度;  $\Delta\tau$ 为信息素浓度的增加值。

## 2.3 改进蚁群算法

在路径选择方面, 传统的蚁群算法既耗时又效率低, 文中提出了一种改进的蚁群算法, 该算法综合考虑了时间、负载和能耗约束等。

### (1) 系统初始化

在云数据中心, 计算资源性能参数被视为节点信息素, 包括: CPU速度 $P_j$  (MIPS)、CPU数 $num_j$ 、内存容量 $R_j$  (MB)、带宽大小 $W_j$  (bit)。首先, 通过式(8)对所有虚拟机信息素进行初始化。其中,  $b_{cpu}$ 、 $b_{mem}$ 、 $b_{net}$ 为模型的特定常数, 在 $k$ 台虚拟机上随机放置 $k$ 只蚂蚁<sup>[21]</sup>。

$$\tau_j(0) = b_{cpu}(num_j \times P_j) + b_{mem}R_j + b_{net}W_j \quad (8)$$

### (2) 状态转移概率

在迭代期间, 状态转移概率 $P_{ij}^k(t')$ 基于路径上的信息(虚拟机的处理能力和能耗信息)和现有的启发信息来确定。最后, 在时间 $t'$ 第 $k$ 只蚂蚁选择在虚拟机 $VM_j$ 中运行任务 $i$ 的概率如式(9)所示<sup>[22]</sup>。

$$P_j^k(t') = \begin{cases} \frac{c_j \tau_j(t')^\beta}{\sum_{i=1}^n c_i \tau_i(t')^\beta}, & i, j = 1, \dots, n \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

式中:  $\beta$ 为信息的启发式因子, 表明资源信息的重要性;  $\tau_j(t')$ 为 $t'$ 时刻虚拟机 $VM_j$ 的信息素浓度;  $c_j$ 为下一个任务在虚拟机 $VM_j$ 上预期执行程度。

$c_j = I_j(t)$  是一个收益函数, 表示要在虚拟机  $VM_j$  上执行的下一个任务的收益, 如式(10)所示。

$$I_j(t) = \frac{1}{\lambda T_j(t)} - \lambda' E_j(t) \quad (10)$$

式中,  $\lambda$  和  $\lambda'$  为经验参数, 使用文中的算法进行计算, 并根据实验结果调整参数。当虚拟机  $j$  被选中或故障时  $P_j^k(t') = 0$ ;  $E_j(t)$  和  $T_j(t)$  分别为下一个任务在虚拟机  $j$  中的能耗和执行时间。

(3) 信息素

通过局部更新对节点上信息素进行动态修正, 所有蚂蚁完成路径搜索后, 更改信息素浓度如式(11)所示<sup>[23]</sup>。

$$\tau_i(t'+1) = (1 - \sigma)\tau_i(t') + \Delta\tau_j \quad (11)$$

式中:  $\tau_i(t'+1)$  为  $t'+1$  时刻虚拟机  $VM_j$  的信息素浓度;  $\Delta\tau_j$  计算如式(12)所示<sup>[24]</sup>。

$$\Delta\tau_j = D \cdot \text{MAX}(\text{RES}(I_{k,i})) \quad (12)$$

式中:  $D$  为信息素强度;  $\text{RES}(I_{k,i})$  为第  $i$  只蚂蚁在第  $k$  次迭代中的分配方案, 通过收益函数计算。

当蚂蚁  $k$  找到分配方案, 路径上所有虚拟机对本地信息素进行更新。找到最优路径, 对局部信息素进行更新。改进蚁群算法的步骤如下:

- 步骤 1) 获取主机状态信息并调控主机电压;
- 步骤 2) 获取可用的虚拟机序列<sup>[25]</sup>;
- 步骤 3) 计算收益矩阵  $I_j(t)$ ;
- 步骤 4) 为可用计算节点设置初始信息素;
- 步骤 5) 初始化蚁群, 随机分布在虚拟机中;
- 步骤 6) 计算蚂蚁  $k$  的转移概率, 选择下一跳节点;
- 步骤 7) 在蚂蚁完成当前搜索后更新本地信息素, 否则, 返回上一步。
- 步骤 8) 在所有蚂蚁完成对这一轮的搜索后, 更新全局信息素。否则, 返回到步骤 6)。
- 步骤 9) 当达到迭代次数时, 任务调度模块输出最优分配计划。否则, 返回到步骤 5)。
- 步骤 10) 确定任务队列中是否有任务要处理。如果是, 返回步骤 1)。否则, 任务分配结束。

改进蚁群算法流程如图 4 所示。

### 3 实验结果与分析

#### 3.1 实验参数

通过实验对文中改进方法的准确性和优越性进行验证。实验中使用的计算机是 Intel i5 处理器, 2.3G Hz, 8G 内存, 并使用 CloudSim 进行仿真<sup>[26]</sup>。

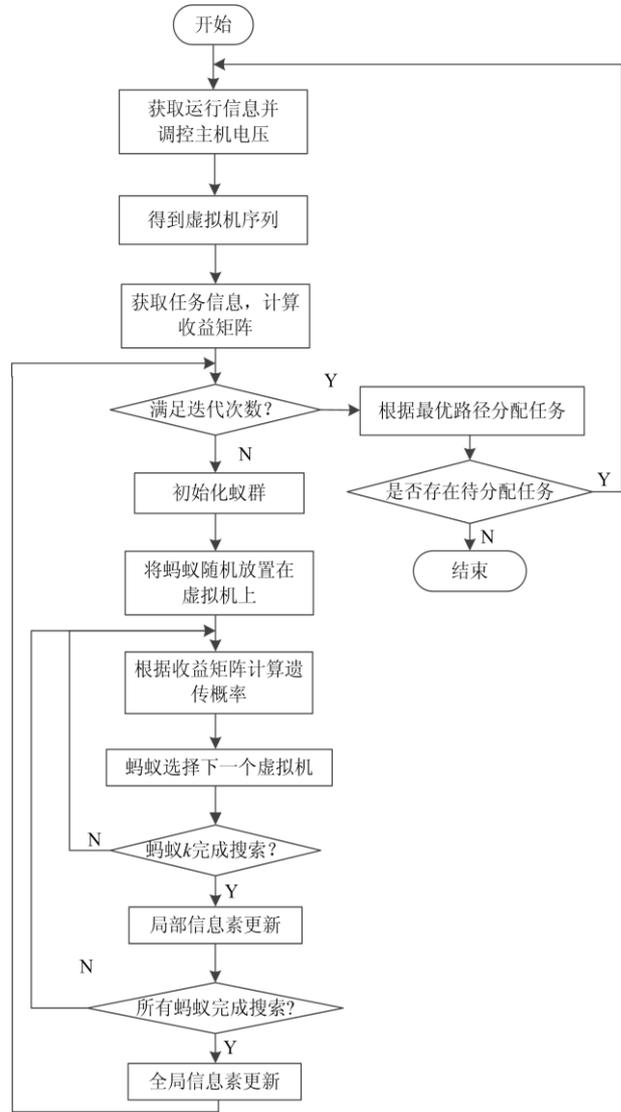


图 4 改进蚁群优化算法流程图

Fig. 4 Improved ant colony optimization algorithm

在实验中, 采用 60 主机搭载 300 台虚拟机, 任务数为 1 000~4 000。具体参数设置见表 1。

实验中, 改进蚁群算法的参数设置如表 2 所示。

表 1 仿真参数

Table 1 Simulation parameters

参数名称	数值
中心数	5
主机数	60
虚拟机数	300
每台虚拟机数	2~8
容量	512~2 048(MB)
带宽	500~1 000(Bit)
任务数	1 000~4 000
任务长度	5 000~10 000(MI)

表 2 算法参数

Table 2 Algorithm parameters

参数名称	蚂蚁数	迭代数	$\alpha$	$\beta$
数值	25	20	0.2	0.5

### 3.2 实验结果与分析

为了验证本文所提算法的优越性, 从四个方面进行分析: 任务执行时间、能耗、任务等待时间和系统负载, 并与改进前的蚁群算法和 Min-Min 算法进行比较<sup>[27]</sup>。

使用文中改进算法和基本蚁群算法、Min-Min 算法, 比较任务数为 1 000~4 000 时的执行时间。实验结果如图 5 所示。横轴是执行任务的数量, 纵轴是任务执行时间。

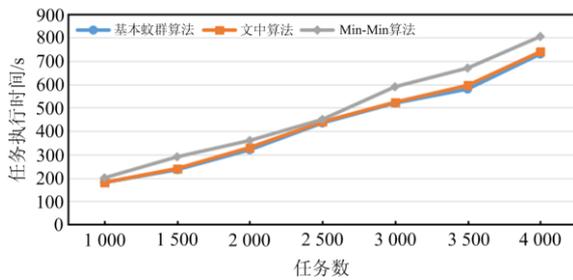


图 5 算法改进前后任务时间对比

Fig. 5 Comparison of task time before and after algorithm improvement

从图 5 中可以看到, 随着任务数量的增加, 任务完成时间也随之增加。Min-Min 算法的任务执行消耗时间最长, 文中算法的结果与基本蚁群算法相似。这是因为文中算法考虑了时间和能耗因素, 而基本蚁群算法只考虑时间系数, 因此从任务执行时间指标上, 基本蚁群算法略优于文中算法。结果表明, 文中算法增加了能耗因素, 但并没有显著增加完成时间。

使用文中算法和基本蚁群算法、Min-Min 算法, 比较任务数为 1 000~4 000 时的能耗。实验结果如图 6 所示。横轴是执行的任务数, 纵轴是机器的能耗。

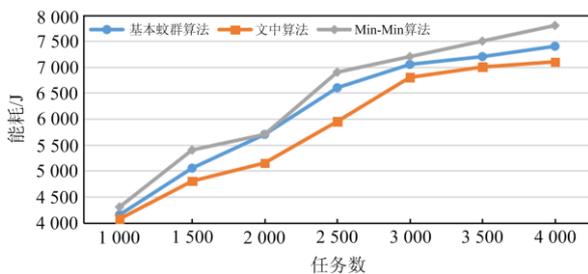


图 6 算法改进前后能耗对比

Fig. 6 Comparison of energy consumption before and after algorithm improvement

从图 6 中可以看到, 随着任务数量的增加, 执行这些任务所消耗的能量也随之增加。Min-Min 算法的任务执行消耗的能量最多, 文中算法消耗能量最少, 与基本蚁群算法相比, 文中算法用于任务调度, 显著降低了系统能耗。这是因为基本蚁群算法和 Min-Min 算法仅关注时间指标, 而不考虑能耗。

使用文中改进算法和基本蚁群算法、Min-Min 算法, 比较任务数为 1 000~4 000 时任务的平均等待时间。实验结果如图 7 所示, 横轴是执行的任务数, 纵轴是平均等待时间。

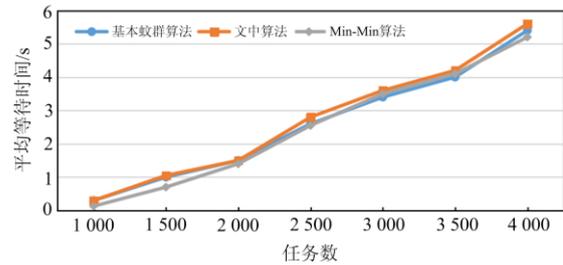


图 7 算法改进前后平均等待时间对比

Fig. 7 Comparison of average waiting time before and after algorithm improvement

从图 7 可以看出, 文中改进算法的任务等待时间略长于基本蚁群算法和 Min-Min 算法的任务等待时间。这是因为文中算法具有较高的时间复杂度, 花费了大量时间计算分配方案所致。

使用文中改进算法和蚁群算法、Min-Min 算法, 对 1 000~4 000 个任务的系统负载进行比较。实验结果如图 8 所示, 横轴是执行的任务数, 纵轴是系统负载。

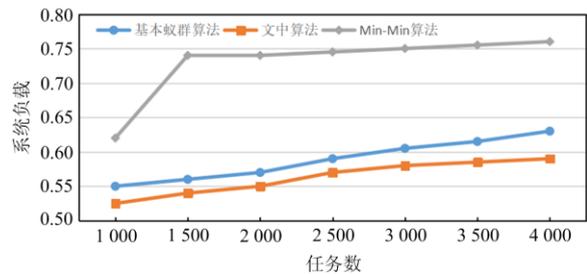


图 8 算法改进前后系统负载对比

Fig. 8 System load comparison before and after algorithm improvement

实验结果表明, 改进算法可以使系统负载均衡在 0.5 到 0.6 之间, 优于 Min-Min 算法和基本蚁群算法。这是由于基本的蚁群算法、Min-Min 算法未添加负载均衡模块, 系统的稳定性较差。

综上所述, 本文提出的改进蚁群算法在保证性

能的前提下,可以有效降低数据中心的能耗。

#### 4 结论

本文对云数据中心任务调度进行建模,综合考虑时间、负荷和能耗等约束对蚁群算法进行改进,并通过改进算法对模型进行求解。结果表明,文中提出的改进蚁群算法在保证性能的前提下,可以有效降低数据中心能耗,在任务数4 000时,相对于改进前能耗降低了4.05%,系统负载降低了6.35%。考虑到现有的实验设备和数据规模,文中仅对调度策略进行了研究,后期可以通过云强大的计算和存储能力,高效地处理和制作三维可视化内容,使显示更加直观。

#### 参考文献

- [1] 张伟晨,熊永新,李程昊,等.基于改进VDCOL的多馈入直流系统连续换相失败抑制及协调恢复[J].电力系统保护与控制,2020,48(13):63-72.  
ZHANG Weichen, XIONG Yongxin, LI Chenghao, et al. Continuous commutation failure suppression and coordinated restoration of multi infeed DC systems based on improved VDCOL[J]. Power System Protection and Control, 2020, 48(13): 63-72.
- [2] 陈友辉,王耀南,肖赞,等.高压线路除冰机器人越障识别定位算法研究[J].计算机工程与应用,2013,49(24):266-270.  
CHEN Youhui, WANG Yaonan, XIAO Zan, et al. Research on obstacle crossing recognition and positioning algorithm of high voltage line deicing robot[J]. Computer Engineering and Application, 2013, 49(24): 266-270.
- [3] 倪志伟,李蓉蓉,方清华,等.基于离散人工蜂群算法的云任务调度优化[J].计算机应用,2016,36(1):107-112,121.  
NI Zhiwei, LI Rongrong, FANG Qinghua, et al. Cloud task scheduling optimization based on discrete artificial bee colony algorithm[J]. Computer Applications, 2016, 36(1): 107-112, 121.
- [4] 周宇,陈江兴,付俊峰,等.基于云计算的电力任务调度优化策略研究[J].电测与仪表,2020,57(13):28-32,39.  
ZHOU Yu, CHEN Jiangxing, FU Junfeng, et al. Research on power task scheduling optimization strategy based on cloud computing[J]. Electrical Measurement & Instrumentation, 2020, 57(13): 28-32, 39.
- [5] 任金霞,黄艺培,钟小康.基于遗传算法的云任务调度改进算法[J].江西理工大学学报,2018,39(6):90-94.  
REN Jinxia, HUANG Yipei, ZHONG Xiaokang. Improved cloud task scheduling algorithm based on genetic algorithm[J]. Journal of Jiangxi University of Technology, 2018, 39(6): 90-94.
- [6] 罗斯宁,王化龙,李弘宇,等.基于改进蚁群算法的云计算用户任务调度算法[J].电信科学,2020,36(2):95-100.  
LUO Sining, WANG Hualong, LI Hongyu, et al. Cloud computing user task scheduling algorithm based on improved ant colony algorithm[J]. Telecom Science, 2020, 36(2): 95-100.
- [7] MATEOS C, PACINI E, GARINO C G. An ACO-inspired algorithm for minimizing weighted flowtime in cloud-based parameter sweep experiments[J]. Advances in Engineering Software, 2013, 56(2): 38-50.
- [8] KANEMITSU H, HANADA M, NAKAZATO H. Clustering-based task scheduling in a large number of heterogeneous processors[J]. IEEE Transactions on Parallel & Distributed Systems, 2016, 27(11): 3144-3157.
- [9] 汪亚斌,丁峰,郭成昊.面向边缘计算的战术时敏任务调度方法[J].指挥信息系统与技术,2018,9(6):36-40.  
WANG Yabin, DING Feng, GUO Chenghao. Edge computing oriented tactical time sensitive task scheduling method[J]. Command Information System and Technology, 2018, 9(6): 36-40.
- [10] 詹文翰,王瑾,朱清新,等.移动边缘计算中基于深度强化学习的计算卸载调度方法[J].计算机应用研究,2021,38(1):241-245,263.  
ZHAN Wenhuan, WANG Jin, ZHU Qingxin, et al. Computational unloading scheduling method based on deep reinforcement learning in mobile edge computing[J]. Computer Application Research, 2021, 38(1): 241-245, 263.
- [11] 陈磊,何慧雯,王磊,等.基于限流器与断路器协调的混合直流输电系统故障隔离方法[J].电力系统保护与控制,2020,48(19):119-127.  
CHEN Lei, HE Huiwen, WANG Lei, et al. Fault isolation method for hybrid HVDC system based on coordination of current limiter and circuit breaker[J]. Power System Protection and Control, 2020, 48(19): 119-127.
- [12] 王波,张晓磊.基于粒子群遗传算法的云计算任务调度研究[J].计算机工程与应用,2015,51(6):84-88.  
WANG Bo, ZHANG Xiaolei. Research on cloud computing task scheduling based on particle swarm genetic algorithm[J]. Computer Engineering and Application, 2015, 51(6): 84-88.
- [13] 魏赞,陈元元.基于改进蚁群算法的云计算任务调度模型[J].计算机工程,2015,41(2):12-16.  
WEI Fu, CHEN Yuanyuan. Cloud computing task

- scheduling model based on improved ant colony algorithm[J]. *Computer Engineering*, 2015, 41(2): 12-16.
- [14] 张龙信, 周立前, 文鸿, 等. 基于异构云计算的成本约束下的工作流能量高效调度算法[J]. *计算机科学*, 2020, 47(8): 112-118.  
ZHANG Longxin, ZHOU Liqian, WEN Hong, et al. Energy efficient scheduling algorithm of workflow under cost constraint based on heterogeneous cloud computing[J]. *Computer Science*, 2020, 47(8): 112-118.
- [15] PATEL G K, DABHI V K, PRAJAPATI H B. Clustering using a combination of particle swarm optimization and K-means[J]. *Journal of Intelligent Systems*, 2017, 12(3): 457-469.
- [16] GAUTAM J V, PRAJAPATI H B, DABHI V K, et al. Empirical study of job scheduling algorithms in Hadoop MapReduce[J]. *Cybernetics and Information Technologies*, 2017, 21(1): 146-163.
- [17] CAETANO C E F, LIMA A B, PAULINO J O S, et al. A conductor arrangement that overcomes the effective length issue in transmission line grounding[J]. *Electric Power Systems Research*, 2018, 46(5): 159-162.
- [18] HU Jianjiang, FICHTNER M, BARICCO M. Preparation of Li-Mg-N-H hydrogen storage materials for an auxiliary power unit[J]. *International Journal of Hydrogen Energy*, 2017, 42(27): 17144-17148.
- [19] JIA Zhiwei, WANG Lijun, ZHANG Jinchuan, et al. High efficiency, low power-consumption DFB quantum cascade lasers without lateral regrowth[J]. *Nanoscale Research Letters*, 2017, 12(1): 88-95.
- [20] 张婕, 曾国辉, 赵晋斌, 等. 基于改进冒泡排序的模块化多电平换流器电容电压均衡策略[J]. *电力系统保护与控制*, 2020, 48(6): 92-99.  
ZHANG Jie, ZENG Guohui, ZHAO Jinbin, et al. Capacitor voltage equalization strategy for modular multilevel converter based on improved bubble sorting[J]. *Power System Protection and Control*, 2020, 48(6): 92-99.
- [21] 陈驰, 彭向阳, 宋爽, 等. 大型无人机电力巡检 LiDAR 点云安全距离诊断方法[J]. *电网技术*, 2017, 41(8): 2723-2730.  
CHEN Chi, PENG Xiangyang, SONG Shuang, et al. Safety distance diagnosis of large scale transmission line corridor inspection based on LiDAR point cloud collected with UAV[J]. *Power System Technology*, 2017, 41(8): 2723-2730.
- [22] 孙立明, 杨博. 蓄电池/超导混合储能系统非线性鲁棒分数阶控制[J]. *电力系统保护与控制*, 2020, 48(22): 76-83.  
SUN Liming, YANG Bo. Nonlinear robust fractional order control for battery / superconducting hybrid energy storage systems[J]. *Power System Protection and Control*, 2020, 48(22): 76-83.
- [23] 戴志辉, 黄敏, 苏怀波. 基于 MMC 的环状直流配网在不同接地方式下的故障特性分析[J]. *电力系统保护与控制*, 2019, 47(1): 1-10.  
DAI Zhihui, HUANG Min, SU Huaibo. Fault characteristics analysis of circular DC distribution network under different grounding modes based on MMC[J]. *Power System Protection and Control*, 2019, 47(1): 1-10.
- [24] 王利平, 庞晓艳, 朱雨, 等. 基于物联网和移动互联的二次设备运维技术研究与应用[J]. *中国电力*, 2019, 52(3): 177-184.  
WANG Liping, PANG Xiaoyan, ZHU Yu, et al. Research and application of secondary equipment operation and maintenance technology based on Internet of Things and mobile interconnection[J]. *Electric Power*, 2019, 52(3): 177-184.
- [25] 盛津芳, 滕潇雨, 李伟民, 等. 移动边缘计算中基于改进拍卖模型的计算卸载策略[J]. *计算机应用研究*, 2020, 37(6): 1688-1692.  
SHENG Jinfang, TENG Xiaoyu, LI Weimin, et al. Computing offload strategy based on improved auction model in mobile edge computing[J]. *Computer Application Research*, 2020, 37(6): 1688-1692.
- [26] 王德文, 刘晓萌. 基于虚拟机动态迁移的电力仿真云计算平台资源调度策略[J]. *计算机科学*, 2015, 39(12): 97-105.  
WANG Dewen, LIU Xiaomeng. Resource scheduling strategy of power simulation cloud computing platform based on virtual machine dynamic migration[J]. *Computer Science*, 2015, 39(12): 97-105.
- [27] BRAUN T D, SIEGEL H J, BECK N, et al. A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogeneous distributed computing systems[J]. *Journal of Parallel & Distributed Computing*, 2001, 61(6): 810-837.

收稿日期: 2021-04-22; 修回日期: 2021-06-05

作者简介:

孙湛冬(1989—), 男, 博士, 高级工程师, 从事电网规划设计工作; E-mail: sun250072@163.com

焦娇(1988—), 女, 博士, 高级工程师, 从事电网规划设计工作;

李伟(1977—), 男, 硕士, 教授级高工, 从事电网规划设计工作。

(编辑 张爱琴)