

基于改进多目标布谷鸟搜索的资源调度算法

程曦¹ 宋铁成²

¹(四川文理学院康养产业学院 四川 达州 635000)

²(重庆邮电大学通信与信息工程学院 重庆 400065)

摘要 针对 IaaS (Infrastructure as a Service) 云计算中资源调度的多目标优化问题,提出一种基于改进多目标布谷鸟搜索的资源调度算法。在多目标布谷鸟搜索算法的基础上,通过改进随机游走策略和丢弃概率策略提高了算法的局部搜索能力和收敛速度。以最大限度地减少完成时间和成本为主要目标,将任务分配特定的 VM (Virtual Manufacturing) 满足云用户对云提供商的资源利用的需求,从而减少延迟,提高资源利用率和服务质量。实验结果表明,该算法可以有效地解决 IaaS 云计算环境中资源调度的多目标问题,与其他算法相比,具有一定的优势。

关键词 云计算 资源调度 多目标布谷鸟搜索 多目标优化

中图分类号 TP393 文献标志码 A DOI:10.3969/j.issn.1000-386x.2022.03.039

RESOURCE SCHEDULING ALGORITHM BASED ON IMPROVED MULTI-OBJECTIVE CUCKOO SEARCH

Cheng Xi¹ Song Tiecheng²

¹(School of Health Caring Industry, Sichuan University of Arts and Science, Dazhou 635000, Sichuan, China)

²(School of Communications and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract Aiming at the multi-objective optimization problem of resource scheduling in IaaS cloud computing, a resource scheduling algorithm based on improved multi-objective cuckoo search is proposed. Based on the multi-objective cuckoo search algorithm, the algorithm improved the local search ability and convergence speed by improving the random walk strategy and discard probability strategy. The main objective was to minimize the makespan time and cost, and the assign tasks to specific VMs to meet the needs of cloud users for resource utilization by cloud providers, thereby reducing latency, improving resource utilization and service quality. The experimental results show that the proposed algorithm can effectively solve the multi-objective problem of resource scheduling in the IaaS cloud computing environment, and has certain advantages compared with other algorithms.

Keywords Cloud computing Resource scheduling Multi-objective cuckoo search Multi-objective optimization

0 引言

云计算是一种通过网络为用户提供软件、数据库、计算、存储和安全性服务的范例。资源管理是云计算中提出的核心问题,包括资源分配、资源调度、资源发现、资源监视、资源可用性和资源定价等方面^[1-2]。资源

调度是从适当的物理资源中选择最佳的虚拟化资源,即对要在其中生成虚拟机 (VM) 的物理资源进行分类,以分配来自云设备的资源^[3]。因此,在云计算环境中的基础架构即服务 (IaaS) 下,资源调度是一个多目标问题^[4]。为此,需要一种优化算法来解决多目标问题,实现资源合理调度。

多目标优化的任务是针对一组确定的限制同时优

化两个或多个冲突目标。但是,在云计算环境下,优化过程容易出现改善一个目标会导致另一个目标降级的难题。近年来,研究人员采用元启发式优化算法来处理资源调度的多目标问题^[5]。Gobalakrishnan 等^[6]提出了一种基于遗传算法和灰狼优化算法相结合的优化技术,实现了云计算环境下负荷利用率、能耗、迁移成本和迁移时间组成的多目标函数的优化,实现了高效的任务调度,降低了时间和迁移成本,但是该算法存在求解精度低和后期收敛速度慢等问题。Agarwal 等^[7]提出一种基于布谷鸟搜索的任务调度方法,该方法在可用的虚拟机之间有效地分配任务,并保持总体响应时间最小,使计算资源得到最佳的利用。Krishnadoss 等^[8]提出了一种基于布谷鸟搜索算法和反向学习算法的多目标任务调度策略,该策略采用完成时间和成本作为优化问题的重要约束条件来求解任务调度的 NP 完全问题,实现了高性能、低成本的资源动态分配的目标。虽然布谷鸟搜索算法操作简单、通用性强,但是存在搜索速度慢、容易陷入局部最优的缺点。Srichandan 等^[9]通过结合遗传算法和细菌觅食算法两种生物启发式算法从经济和生态的角度方面降低了能源消耗。Abdullahi 等^[10]针对 IaaS 云计算环境下的多目标大规模任务调度优化问题,提出了一种混沌共生生物搜索算法。采用混沌优化策略生成初始种群,并用混沌序列代替基于随机序列的 SOS 相分量,以保证生物多样性,实现了全局收敛,但是收敛速度慢。

针对当前用于解决 IaaS 云计算的资源调度多目标优化算法中存在的诸多问题,提出一种基于改进多目标布谷鸟搜索的资源调度算法,通过改进随机游走策略和丢弃概率策略提高了算法的局部搜索能力和收敛速度,从而实现以最低的执行时间和成本将任务分配特定的 VM 中,满足云用户对云提供高的资源利用需求,减少延迟,提高资源利用率和服务质量。

1 云计算资源调度描述

云计算中的资源调度是一种多目标优化过程^[11],适用于处理器、网络、存储和 VM 等云资源的分发,根据云用户的需求平均分配资源,实现云资源的最佳利用,确保云计算能够以云提供商所提供的一定质量的服务来满足所有云用户的请求。资源调度(Resource Scheduling, RS)问题可以描述为:

$$RS = \sum_{x=1}^{m,n} (R_x + S_x + \dots + N_x) \times T_x \rightarrow U_x^z \quad (1)$$

式中:将 m 个云任务 T_x 映射到 n 个任务云数据中心虚拟资源上,如 R_x 、 S_x 和 N_x ; U_x^z 表示云用户。

假设存在一组任务 $T_i = (T_1, T_2, \dots, T_n)$,云计算已将其映射到虚拟资源 $V_j = (V_1, V_2, \dots, V_m)$ 上,而后被调度到物理设备中执行。则任务与资源之间的对应关系可以用矩阵表示为:

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{bmatrix} \quad (2)$$

式中: $x_{ij} \in \{0,1\}$ 表示为第 i 个任务 T_i 与第 j 个资源 V_j 的对应关系,当 $x_{ij} = 1$ 时,表示任务 T_i 占据资源 V_j ,反之则没有。根据任务与资源之间的对应关系,任务 T_i 经过资源传递到达物理设备上执行的预期完成时间可以表示为:

$$ETC(T_i, V_j) = \begin{bmatrix} T_1 V_1 & T_1 V_2 & \dots & T_1 V_m \\ T_2 V_1 & T_2 V_2 & \dots & T_2 V_m \\ \vdots & \vdots & & \vdots \\ T_n V_1 & T_n V_2 & \dots & T_n V_m \end{bmatrix} \quad (3)$$

同理,预期完成成本 ECC 可以定义为:

$$ECC(C_i, V_j) = \begin{bmatrix} C_1 V_1 & C_1 V_2 & \dots & C_1 V_m \\ C_2 V_1 & C_2 V_2 & \dots & C_2 V_m \\ \vdots & \vdots & & \vdots \\ C_n V_1 & C_n V_2 & \dots & C_n V_m \end{bmatrix} \quad (4)$$

式中: $C_i = (C_1, C_2, \dots, C_n)$ 表示为任务成本。

根据预期完成时间 ETC 和预期完成成本 ECC,云计算环境中资源节点完成子任务所花费的时间和成本可以计算为:

$$sumTime(i) = \max \sum_{i=1}^n \sum_{j=1}^m t(i,j) \times ETC(i,j) \quad (5)$$

$$sumCost(i,j) = \sum_{i=1}^n \sum_{j=1}^m res(i,j) \times ECC(i,j) \quad (6)$$

式中: $t(i,j)$ 和 $res(i,j)$ 分别表示任务 T_i 在资源 V_j 的所需时间和成本; $sumTime(i,j)$ 和 $sumCost(i,j)$ 分别表示任务完成的总时间和总成本。

2 改进多目标布谷鸟搜索的调度策略

当前大多数资源调度算法通过考虑不同的因素(如完成时间、执行所有用户任务的总成本、资源利用

率、能耗和容错能力)进行优化,优先约束并行应用的求解时间与能耗之间的折中问题是一个双目标优化问题。这个问题的解决办法是一组帕累托点。针对大多数情况下 IaaS 云资源调度不够而产生的利用率低的问题,本文提出一种基于改进多目标布谷鸟搜索的资源调度算法,以完成时间和成本为优化目标,用最小的 ETC 和 ECC 矩阵将任务分配到虚拟资源上,提高资源利用率。

2.1 多目标布谷鸟搜索算法及改进措施

布谷鸟搜索算法(Cuckoo Search Algorithm, CSA)^[12]是在 2010 年由 Yang 等提出的一种生物启发式智能优化算法,该算法参考了自然界中布谷鸟的寄宿繁衍行为和果蝇的莱维飞行行为,结构简单,易于实现,具有很好的全局搜索能力。鉴于 CSA 局限于单目标问题的优化, Yang 等在 2013 年提出了多目标布谷鸟搜索优化(Multi-Objective Cuckoo Search Optimization, MOCSO)算法^[13],该算法可以直接求解 Pareto 最优解集,应用于多目标优化领域。

多目标布谷鸟搜索算法的 3 个基本假设是在原来 CSA 假设的基础上为满足 k 个目标的需求而做出一定的修改:

- (1) 每只布谷鸟一次可产 k 个蛋,并随机选择一个寄生巢放置,第 k 个蛋即是一组解的第 k 个目标。
- (2) 在随机选择的一组寄生巢中,最好的巢将会保留到下一代继续繁殖。
- (3) 每个巢中的宿主鸟丢弃外来蛋的概率为 P_a ,被发现后布谷鸟选择更换一个具有 k 个蛋的新巢。

布谷鸟蛋的寄生巢表示搜索空间的一个解,寄生巢位置表示解的适应度值,布谷鸟搜索可以通过莱维飞行来更新 $t + 1$ 时刻的位置:

$$x_i^{t+1} = x_i^t + \alpha \cdot Levy(\beta) \quad (7)$$

式中: $i = 1, 2, \dots, n$, $Levy$ 为莱维飞行搜索; α 为步长控制量,可以引入不同解之间的差来增加算法的收敛速度。 α 的计算如下:

$$\alpha = \alpha_0 (x_j^t - x_i^t) \quad (8)$$

式中: α_0 是个常数。

MOCSO 中鸟巢位置的更新由解的相似性决定的:

$$x_i^{t+1} = x_i^t + r(x_k^t - x_j^t) \quad (9)$$

式中: r 为缩放因子; x_k^t 和 x_j^t 是从解集中随机选中的两个解。

图 1 给出了多目标布谷鸟搜索算法的流程。

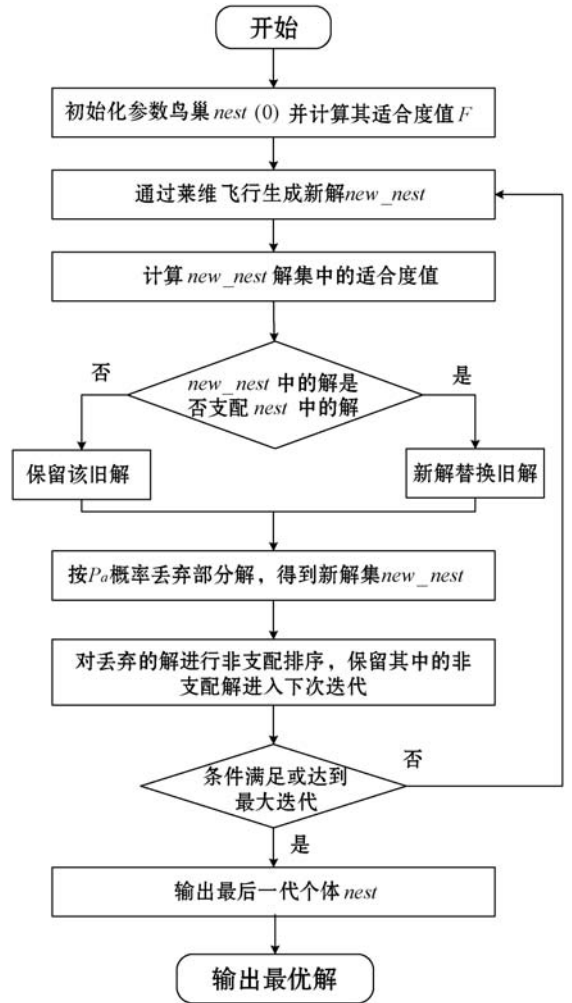


图 1 多目标布谷鸟搜索算法流程示意图

在多目标优化过程中,其理想最优解是可以完全支配其他解的解,但在优化过程中往往获得的是多个相互妥协的解,即 Pareto 解。虽然 MOCSO 考虑了新解和旧解之间的支配关系,但是由于该算法是基于单目标算法之上的,在优化过程中缺乏对个体之间相互支配关系的考虑。除此之外,MOCSO 中的游走策略虽然能够很好地保持算法的全局搜索能力和解的多样性,但是由于缩放因子 r 是一个随步长而改变的随机数,在搜索后期会因步长的变小而变小,使得解的多样性降低,陷入局部最优。同时,发现概率 P_a 为一固定值,无论解的适合度值多高,均会存在 P_a 概率被丢弃,这种方式会忽略较好的解,影响最终优化结果,也不适用于 MOCSO。

针对上述两点,本文从丢弃频率和游走策略两个方面做适当改进。首先重新定义丢弃概率:将优化过程中产生的新解与旧解合并,按适应度大小排序;修改丢弃概率 P_a ,将其设定为一个区间 $P_a \in [P_{min}, P_{max}]$;最后将区间 $[P_{min}, P_{max}]$ 划分为等间距且长度与解个数相同的子集,即形成一一对应关系,最高适应度的解对应最低丢弃概率,反之对应最高丢弃概率。而丢弃概

率的最大值与最小值定义为:

$$P_{\max} = \max(P_{\max}) \cdot (1 - t/\max_iter) \quad (10)$$

$$P_{\min} = \max(P_{\min}) \cdot (1 - t/\max_iter) \quad (11)$$

式中: \max_iter 表示最大迭代次数。

其次,针对式(9)中存在的搜索后期解空间内多样性降低的问题,提出下面的游走策略:

$$x_i^{t+1} = x_i^t + rand \times (r \times x_i^t - (1 - r) \times x_j^t) \quad (12)$$

式中: $rand$ 是在均匀分布在 $[0, 1]$ 中的随机数,该策略增加了解更新方向的随机性,从而增强了解空间个体的多样性。

2.2 多目标资源调度问题

本文将完成时间和成本作为 *IaaS* 云计算环境中优化资源调度的多目标函数,利用改进的 *MOCSO* 对多目标函数进行优化。云计算资源调度的目标适应度函数是花费时间和成本的加权:

$$F = \min(\lambda \cdot sumTime + \mu \cdot sumCost) \quad (13)$$

式中: λ 和 μ 表示加权系数。基于改进的 *MOCSO* 的多目标资源调度算法的伪代码由算法 1 给出。

算法 1 基于改进 *MOCSO* 算法的多目标资源调度算法

输入: 丢弃概率 $P_a \in [P_{\min}, P_{\max}]$, 游走策略参数 r , 种群数量为 n 的初始化 $x_i (i = 1, 2, \dots, n)$, 最大迭代 Max_{iter} , 目标函数 $F_i(x) = f(x_i)$ 。

输出: 最优解 S_{best}

```

While(( $t < Max_{iter}$ ) or (停止条件)) do
    基于莱维飞行随机生成一个解; //新解  $x_i^{t+1}$ 
    评估解的适合度  $F_i$ ; //  $F_i = f(x_i^{t+1})$ 
    if ( $F_i$  是 Pareto 最优)
        从  $n$  个巢中随机选择一个  $j$  巢; //旧解  $x_j^t$ 
        评估  $j$  巢的  $K$  个解;
        if ( $F_i > F_j$ ) then
            将  $F_i$  替换  $F_j$ ;
        end if
    end if
    丢弃部分适合度低的巢,通过莱维飞行在新位置建造新巢;保留最好的巢,留给下一代;
    对当前解进行排序并找到当前最佳的 Pareto 最优;
     $t \leftarrow t + 1$ ;
end while
返回最优解  $S_{best}$ ;

```

3 实验与结果分析

为了评价本文算法的性能,使用 CloudSim3.0 仿真软件进行测试实验,并在相同条件下与基于布谷鸟

多目标优化(MOCSO)、基于简化群多目标优化(MOSSO)^[14]、基于粒子群多目标优化(MOPSO)^[15]和基于蚁群多目标优化算法(MOACO)^[16]的任务调度方法相比。采用完成时间、成本和利用率三个性能评价指标评估本文方法的性能。表 1 给出了仿真实验运行环境的详细信息。

表 1 实验运行环境的详细信息

条目	参数	值
用户	用户数量	100
任务	任务数量	200 ~ 2 000
	长度/m	800 000
	文件大小/B	600
主机	RAM/GB	4
	存储/MB	1 000 000
	带宽/(bit · s ⁻¹)	10 000
虚拟机 VM	VM 数量	50
	RAM/MB	512
	带宽/(bit · s ⁻¹)	10 000
	MIPS	1 000
	VMM	Xen
	操作系统	Linux
数据中心	数据中心数量	2

3.1 数据集及评价指标

本文采用 HPC2N、NASA Ames iPCS / 860 和 SDSC 三个工作负载档案集^[17]进行测试,这些档案集提供了 CloudSim 工具认可的标准工作负载格式(.swf)。HPC2N 数据集包括 527 371 个任务的统计数据,NASA 包括 14 794 个任务的统计数据,SDSC 包括 73 496 个任务的统计数据。在云计算环境中,这些数据集用于评估算法的性能。

为了对比本文算法与其他启发式资源调度优化算法的测试结果,采用完成时间、成本、资源利用率三个指标来评进行价。下面给出三个指标的数学定义。

资源调度的完成时间是通过 VM 上所有任务映射的完成时间来确定执行的最大完成时间:

$$f(x) = \max_{i=1}^m t_i \quad (14)$$

式中: t_i 表示任务 T_i 的完成时间。

成本表示针对资源使用或利用率产生的总金额,VM 的成本根据云提供商所确定的大量时间和 VM 的描述而有所不同。下面给出用于计算完成特定 VM 任务的成本:

$$f(y) = \sum_{i=1}^m resource^i (C_i \times t_i) \quad (15)$$

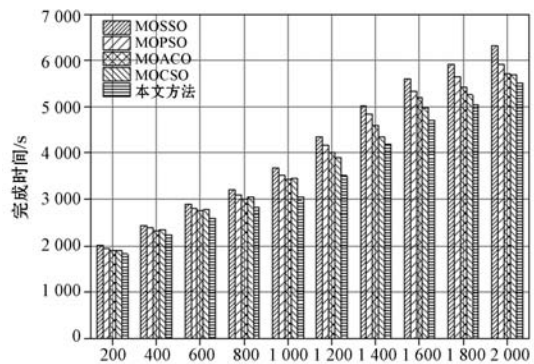
式中: C_i 表示单位时间内资源 i 的使用成本; $resource^i$ 表示资源 i 。

利用率是指数据中心主映射有效利用的资源量:

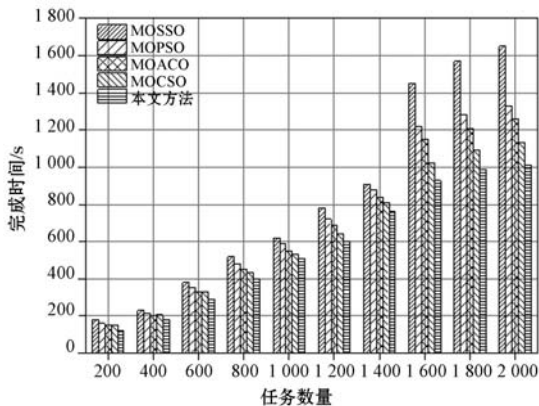
$$f(z) = \frac{\sum_{i=1}^m resource^i(t_i)}{\max \bigcup_{i=1}^m t_i} \quad (16)$$

3.2 结果分析

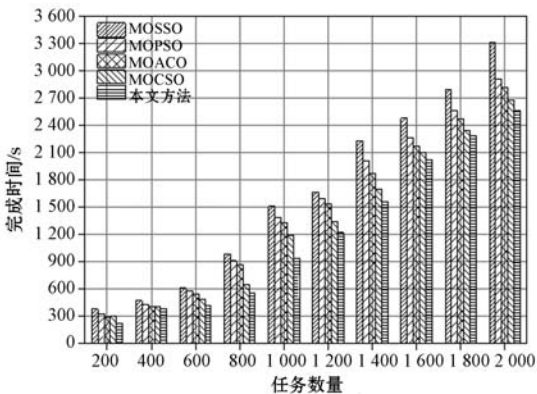
利用多目标资源调度的完成时间、成本和资源利用率三个指标评估提出的改进 MOCSO 在 IaaS 云计算环境中的优化性能。图 2 给出了不同资源调度算法在 HPC2N、NASA 和 SDCS 三个工作负载数据集测试的完成时间。



(a) HPC2N数据集



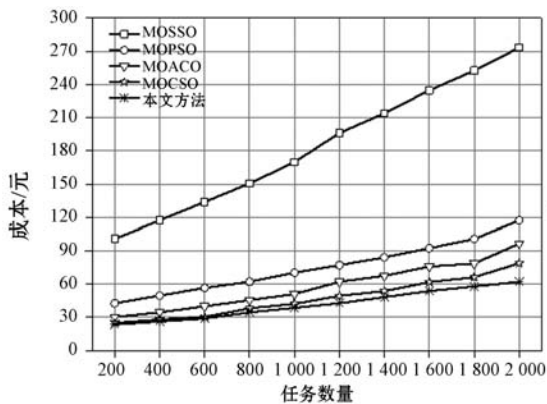
(b) NASA数据集



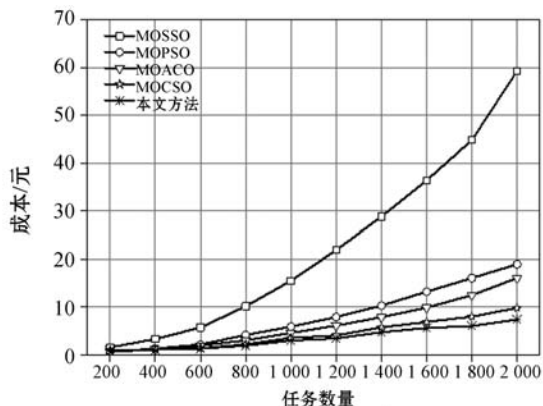
(c) SDCS数据集

测试过程中使用 200 ~ 2 000 范围内各种数量的任务进行仿真。将三种云计算资源调度算法用于与提出的改进 MOCSO 进行比较。可以看出,随着任务数量的增加,资源调度算法的完成时间会增加。结果表明,相比于标准 MOCSO,改进的 MOCSO 的完成时间有所降低,同时还可以看出,本文算法的效果比其他三种对比算法要好。

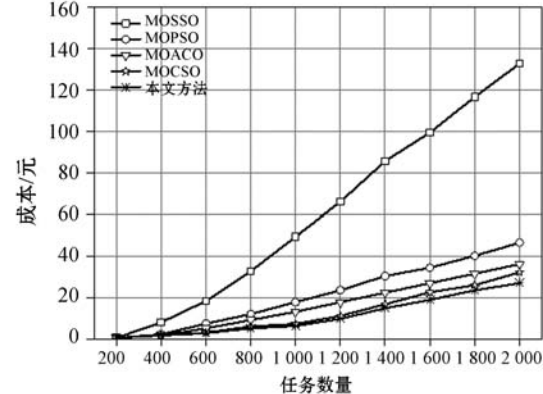
图 3 给出了不同资源调度算法在 HPC2N、NASA 和 SDCS 三个工作负载数据集测试的成本。可以看出,随着任务数量的增加,资源调度算法的成本也随之增加。本文算法计算出的成本低于其他四种算法,该结果表明提出的改进 MOCSO 在云计算环境中支持云用户减少开支。



(a) HPC2N数据集



(b) NASA数据集



(c) SDCS数据集

图 2 不同资源调度算法在三个数据集上的完成时间

图 3 不同资源调度算法在三个数据集上的成本

图 4 给出了不同资源调度算法在 HPC2N、NASA 和 SDCS 三个工作负载数据集测试的资源利用率。可以明显看出,随着任务数量的增加,资源调度算法的资源利用率下降。与其他四种算法相比,本文算法在资源利用率上具有一定的优势。

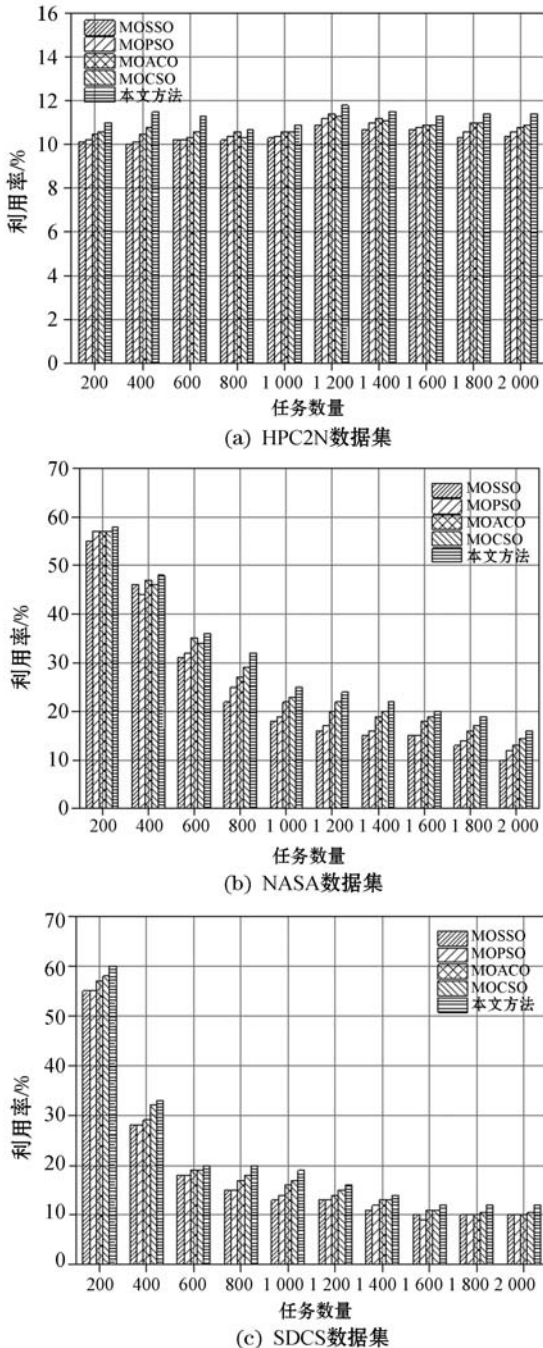


图 4 不同资源调度算法在三个数据集上的利用率

仿真结果和分析结果表明,提出的改进 MOCSO 在完成时间、成本、利用率方面提供了更好的质量结果,相较于标准 MOCSO 有了不错的提升。进一步阐明了改进 MOCSO 非常适合 IaaS 云计算环境中的 ETC 和 ECC 矩阵。从性能评估可以看出,改进 MOCSO 可能是一种强大的搜索和优化技术,足以解决 IaaS 云计算环境中资源调度的多目标问题。

4 结 语

本文提出一种基于改进多目标布谷鸟搜索的资源调度算法,用于解决 IaaS 云计算环境中资源调度的多目标问题。该算法在多目标布谷鸟搜索算法的基础上,为了提高优化算法的局部搜索能力和收敛速度,在随机游走和丢弃概率两个地方做出了一定的改进。本文算法用最小的 ETC 和 ECC 矩阵将任务分配到虚拟资源上,以最大限度地减少完成时间和成本,解决 IaaS 云资源因为调度不够而导致利用率低的问题。实验结果表明,与其他算法相比,本文算法在评估指标上提供了更好的质量结果。

参 考 文 献

- [1] Kang S, Veeravalli B, Aung K M M. Dynamic scheduling strategy with efficient node availability prediction for handling divisible loads in multi-cloud systems[J]. *Journal of Parallel and Distributed Computing*, 2018, 113(1): 1-16.
- [2] 徐健锐,朱会娟. 云计算环境中面向 DAG 任务的多目标调度算法[J]. *计算机应用研究*, 2019, 36(1): 37-42.
- [3] Elaziz M A, Xiong S, Jayasena K P N, et al. Task scheduling in cloud computing based on hybrid moth search algorithm and differential evolution[J]. *Knowledge-Based Systems*, 2019, 169(1): 39-52.
- [4] Madni S H H, Latiff M S A, Coulibaly Y. Recent advancements in resource allocation techniques for cloud computing environment: A systematic review[J]. *Cluster Computing*, 2017, 20(3): 2489-2533.
- [5] Bindu G B H, Ramani K, Bindu C S. Energy aware multi objective genetic algorithm for task scheduling in cloud computing[J]. *International Journal of Internet Protocol Technology*, 2018, 11(4): 242-249.
- [6] Gopalakrishnan N, Arun C. A new multi-objective optimal programming model for task scheduling using genetic gray wolf optimization in cloud computing[J]. *The Computer Journal*, 2018, 61(10): 1523-1536.
- [7] Agarwal M, Srivastava G M S. A cuckoo search algorithm-based task scheduling in cloud computing[M]//*Advances in Computer and Computational Sciences*. Springer, 2018: 293-299.
- [8] Krishnadoss P, Jacob P. OCSA: Task scheduling algorithm in cloud computing environment[J]. *International Journal of Intelligent Engineering and Systems*, 2018, 11(3): 271-279.

- 2014; 33 – 38.
- [2] Jose-Garcia A, Gomez-Flores W. Automatic clustering using nature-inspired metaheuristics: A survey [J]. *Applied Soft Computing*, 2016, 41: 192 – 213.
- [3] Handl J, Knowles J. Evidence accumulation in multiobjective data clustering [C] // *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2013: 543 – 557.
- [4] Babu G P, Murty M N. Clustering with evolution strategies [J]. *Pattern Recognition*, 1994, 27(2): 321 – 329.
- [5] Murthy C A, Chowdhury N. In search of optimal clusters using genetic algorithms [J]. *Pattern Recognition Letters*, 1996, 17(8): 825 – 832.
- [6] Bandyopadhyay S, Maulik U. An evolutionary technique based on K-means algorithm for optimal clustering in RN [J]. *Information Sciences*, 2002, 146(1/4): 221 – 237.
- [7] Handl J, Knowles J. An evolutionary approach to multiobjective clustering [J]. *IEEE Transactions on Evolutionary Computation*, 2007, 11(1): 56 – 76.
- [8] Corne D W, Jerram N R, Knowles J D, et al. PESA-II: Region-based selection in evolutionary multiobjective optimization [C] // *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*. ACM, 2001: 283 – 290.
- [9] Saha S, Bandyopadhyay S. A symmetry based multiobjective clustering technique for automatic evolution of clusters [J]. *Pattern Recognition*, 2010, 43(3): 738 – 751.
- [10] Bandyopadhyay S, Saha S. GAPS: A clustering method using a new point symmetry-based distance measure [J]. *Pattern recognition*, 2007, 40(12): 3430 – 3451.
- [11] Ferreira C. Gene expression programming: A new adaptive algorithm for solving problems [EB]. arXiv: cs/0102027, 2001.
- [12] 陈瑜,唐常杰,叶尚. 基于基因表达式编程的自动聚类方法 [J]. *四川大学学报(工程科学版)*, 2007, 39(6): 107 – 112.
- [13] 姜代红,张三友. 基于基因表达式编程的 K 均值自动聚类算法 [J]. *计算机仿真*, 2010, 27(12): 216 – 220.
- [14] Lin Y, Hong P. Niche gene expression programming based on clustering model [C] // *Proceedings of the Workshop on Intelligent Information Technology Application*. ACM, 2007: 10 – 13.
- [15] 姜代红,尹洪胜,张三友. 采用基因表达式编程的自适应层次聚类方法 [J]. *华侨大学学报(自然科学版)*, 2018, 39(3): 435 – 438.
- [16] 蔡宏果,元昌安. 一种基于基因表达式编程的串行聚类算法并行化研究 [J]. *中南民族大学学报(自然科学版)*, 2017, 36(4): 112 – 115.
- [17] 李婷婷,江朝晖,饶元,等. 结合基因表达式编程与空间模糊聚类的图像分割 [J]. *中国图象图形学报*, 2017, 22(5): 575 – 583.
- [18] 陈超. 基因表达式编程优化算法及其在聚类分析中的应用 [D]. 西安:西安电子科技大学, 2013.
- [19] 姜代红,张三友. 基于基因表达式编程的 K 均值自动聚类算法 [J]. *计算机仿真*, 2010, 27(12): 216 – 220.
- [20] Ni Y, Du X, Xie D, et al. A multi-objective cluster algorithm based on GEP [C] // *International Conference on Cloud Computing & Big Data*. IEEE, 2014.
- [21] Chen Y, Tang C, Zhu J, et al. Clustering without prior knowledge based on gene expression programming [C] // *International Conference on Natural Computation*. IEEE, 2007.
- [22] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II [J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182 – 197.
-
- (上接第 246 页)
- [9] Srichandan S, Kumar T A, Bibhudatta S. Task scheduling for cloud computing using multi-objective hybrid bacteria foraging algorithm [J]. *Future Computing and Informatics Journal*, 2018, 3(2): 210 – 230.
- [10] Abdullahi M, Ngadi M A, Dishing S I, et al. An efficient symbiotic organisms search algorithm with chaotic optimization strategy for multi-objective task scheduling problems in cloud computing environment [J]. *Journal of Network and Computer Applications*, 2019, 133: 60 – 74.
- [11] 王虎,雷建军,万润泽. 基于改进的粒子群优化的云计算资源调度模型 [J]. *华中师范大学学报(自然科学版)*, 2018, 52(6): 788 – 791.
- [12] Liu W, Shi C, Yu H, et al. Task scheduling of an improved cuckoo search algorithm in cloud computing [J]. *International Journal of Performability Engineering*, 2019, 15(7): 1965 – 1975.
- [13] 张鑫狮,刘俊,罗世彬. 基于改进多目标布谷鸟搜索算法的翼型气动优化设计 [J]. *航空学报*, 2019, 40(6): 49 – 62.
- [14] Huang C L, Jiang Y Z, Yin Y, et al. Multi objective scheduling in cloud computing using MOSSO [C] // *2018 IEEE Congress on Evolutionary Computation (CEC)*, 2018.
- [15] Ebadifard F, Babamir S M. A PSO-based task scheduling algorithm improved using a load-balancing technique for the cloud computing environment [J]. *Concurrency and Computation: Practice and Experience*, 2018, 30(12): 1 – 16.
- [16] Reddy G N, Kumar S P. Modified ant colony optimization algorithm for task scheduling in cloud computing systems [M] // *Smart Intelligent Computing and Applications*. Springer, 2019: 357 – 365.
- [17] Vila S, Guirado F, Lerida J L, et al. Energy-saving scheduling on IaaS HPC cloud environments based on a multi-objective genetic algorithm [J]. *The Journal of Supercomputing*, 2019, 75(3): 1483 – 1495.