

基于 CSA 和 PSO 的多目标优化任务调度算法^①

王丽娟¹, 吴 锋²

1. 焦作师范高等专科学校 信息工程学院, 河南 焦作 454000; 2. 信阳农林学院 信息工程学院, 河南 信阳 464006

摘要: 针对现有云计算环境中任务调度算法资源利用率低、完成时间长和调度成本高的问题, 提出一种基于布谷鸟搜索算法(Cuckoo Search Algorithm, CSA)和粒子群优化(Particle Swarm Optimization, PSO)混合的多目标优化任务调度策略. 该策略以完成时间、成本以及最后期限违反率为目标函数, 将布谷鸟搜索算法和粒子群优化有效地组合来执行智能优化任务调度问题, 避免调度过程中出现局部最优现象. 实验结果表明: 相对于其他启发式调度算法, 本文提出的方法明显具有优势, 可以最大程度地降低完成时间、调度成本和最后期限违反率.

关键词: 云计算; 任务调度; 多目标优化; 布谷鸟搜索; 粒子群优化

中图分类号: TP391

文献标志码: A

文章编号: 1000-5471(2021)03-0027-06

云环境是指能够从动态虚拟化的资源池中向用户提供计算能力、存储能力或者虚拟机服务的互联网大数据环境^[1]. 云提供的服务可以分为软件即服务(SaaS)、平台即服务(PaaS)和基础设施即服务(IaaS). 用户可以利用这些服务访问虚拟资源, 但必须以“按次付费”模式为云服务付费. 近年来, 随着云的应用和推广, 分散的云数据中心可以提供更多高质量和可靠的服务. 在使用云服务时, 由于用户在物理资源使用方面直接依赖于云服务, 因此优化任务调度成为一种必要条件^[2-3].

云环境中的任务调度策略是指通过算法为用户任务提供更合理的资源分配方案, 在最短的完成时间内实现资源优化、负载均衡等目标^[4]. 目前, 许多云计算任务调度模型通过引入启发式优化算法来解决任务调度的非确定多项式(Nondeterministic Polynomial, NP)完全问题^[5-6]. 文献[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. 文献[8]为提高云计算任务调度效率, 提出一种新的云计算任务调度算法, 该算法以等待时间、资源负载均衡程度及任务完成所耗费用为目标函数, 利用改进的分数阶达尔文粒子群算法自适应搜索最优解. 文献[9]结合负载利用率、能耗、迁移成本和迁移时间, 提出一种新的多目标函数, 并且将遗传算法和灰狼优化算法相结合对该目标函数进行优化, 有效地改善了任务调度. 文献[10]提出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, 实现高性能、低成本的资源动态分配目标. 文献[11]提出一种混合粒子群算法和爬山算法对任务调度完成时间进行优化, 提高了云计算异构环境中的服务效率.

当综合考虑性能、成本和最后期限违反率时, 上述文献并未提供接近最佳的结果. 针对这一问题, 本文提出一种基于布谷鸟搜索和粒子群优化算法的任务调度策略, 该策略将 CSA 和 PSO 两种算法混合在一起, 混合算法既利用 CSA 算法的搜索随机性, 扩大算法的搜索范围, 又可以利用 PSO 算法的优势提高算法的收敛性, 从而克服 PSO 算法陷入局部最优和 CSA 算法收敛速度慢的缺点, 可以更有效地优化任务和

① 收稿日期: 2020-02-09

基金项目: 河南省科技攻关项目(172102210450).

作者简介: 王丽娟, 硕士, 讲师, 主要从事计算机应用技术研究.

资源, 实现相应虚拟机(Virtual Machine, VM)之间最小的完成时间和成本.

1 相关背景

1.1 布谷鸟搜索算法

布谷鸟搜索算法(Cuckoo Search Algorithm, CSA)^[12]是在 2010 年提出的一种生物启发式智能优化算法, 该算法结构简单, 易于实现, 而且可以与其他优化算法结合, 获得性能更佳的混合算法. CSA 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, 可以为连续优化问题快速有效地寻找到最优解. 该算法全局搜索能力较强, 但容易陷入局部最优.

布谷鸟搜索算法存在 3 个基本假设: ① 每只布谷鸟一次仅产一个卵, 并随机选择一个寄生巢; 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; ③ 可利用寄生巢数量固定, 宿主鸟发现外来卵的概率为 $P_a \in [0, 1]$. 被发现后布谷鸟选择丢弃卵或者更换新巢.

布谷鸟卵寄生的巢即为空间的一个解, 寄生巢位置的好坏代表解的适应度值, 布谷鸟寻找寄生巢的过程就是算法搜索和优化的过程. 布谷鸟寻找寄生巢以及位置 x 更新的公式被定义为

$$x_i(k+1) = x_i(k) + \alpha \cdot Levy(\beta) \quad (1)$$

其中, $i = 1, 2, \dots, n$, α 为步长控制量, $Levy$ 为莱维飞行搜索, 服从 $Levy$ 分布式

$$Levy \sim u = t^{-\beta} \quad (2)$$

1.2 粒子群优化算法

粒子群优化(PSO)^[13]算法是在 1995 年提出的一种基于群体智能的全局随机搜索算法. 在 PSO 中, 每个优化问题的潜在解都是搜索空间中的一个粒子, 所有粒子都有一个被优化的函数决定的适应度值, 每个粒子的飞行速度还决定其飞行的方向和距离, 粒子群追随当前的最优粒子在解空间中搜索.

PSO 初始化为一群随机粒子, 然后通过迭代找到最优解. 在每一次迭代中, 粒子通过跟踪粒子本身所找到的最优解 p_i 和整个种群目前找到的最优解 p_g 两个极值来更新自己的位置和速度. 假设在 d 维的目标搜索空间中, 粒子群包含有 N 个粒子, 则第 i 个粒子在 $k+1$ 时刻的速度和位置可以根据 k 时刻的速度 $v_i^d(k)$ 和位置 $y_i^d(k)$ 更新为

$$v_i^d(k+1) = \omega v_i^d(k) + c_1 r_1 (p_i^d(k) - y_i^d(k)) + c_2 r_2 (p_g^d(k) - y_i^d(k)) \quad (3)$$

$$y_i^d(k+1) = y_i^d(k) + \alpha v_i^d(k+1) \quad (4)$$

其中, $i = 1, 2, \dots, N$, $d = 1, 2, \dots, D$, ω 为惯性权重, c_1, c_2 表示学习因子, $r_1, r_2 \in [0, 1]$ 是随机数, α 为约束因子, 用来控制速度的权重.

2 基于 CSA 和 PSO 的多目标任务调度算法

任务调度的主要目标是有效地将任务分配给云服务提供商的相应虚拟机(VM), 并使得目标函数最小化. 为了改善和优化调度性能和成本, 提高云环境下的资源利用效率, 本文提出一种多目标函数混合任务调度算法, 通过引入布谷鸟搜索和粒子群优化算法对目标问题进行优化. 许多用户将任务提交到任务管理器, 任务管理器的作用是接受任务和管理任务. 同时, 任务管理器通过预算、成本、内存和最后期限提交任务至调度程序. 任务由调度程序进行调度和映射到全局资源管理器. 全局资源管理器全局控制资源, 监视资源上的持续时间任务和计算资源成本. 整个过程在云中完成, 该云允许存在各种物理节点和虚拟机.

在云计算中调度算法的效率取决于性能和预算成本. 在定义任务、资源和成本的基础上, 提出一种多目标优化调度模型, 以优化云中的调度. 提出的优化模型将 K 个任务调度到 N 个资源上, 从而达到优化的目的. 除了优化, 还考虑了最后期限和预算成本的约束条件. 假定每个任务的执行都必须在单个 VM 中完成.

在当前云计算中, 存在 N 个资源 $R = \{R_1, R_2, \dots, R_N\}$ 和 K 个任务 $T = \{T_1, T_2, \dots, T_K\}$. 云上的每个资源由中央处理器(CPU)和内存来定义 $R_j = (C_j; M_j)$, 用户提交的任务定义为 $T_i = (C_i; M_i; DL_i; B_i)$,

其中 C, M 分别表示 CPU 和内存利用率, DL 表示任务的最后期限, B 表示用户预算成本. 多目标优化问题可以描述为

$$Fitness(x) = Minimize \sum_x MakSpan(x), Budget(x) \quad (5)$$

$$MakSpan(x) = \sum_{i=1}^K DL_i \quad (6)$$

$$Budget(x) = \sum_{i=1}^K B_i \quad (7)$$

$$Budget(x) = CPU(x) + Memory(x) \quad (8)$$

其中, 目标函数的性能由 $MakSpan(x)$ 表示, x 表示可行的解决方案. $Budget(x)$ 定义预算成本和对应成本的资源(CPU 和内存). 式(7)和式(8)给出了用户预算与资源成本模型中资源成本之间的关系. 根据资源定义, 通过将资源成本划分为 CPU 和内存来使用资源成本模型. 式(9)、式(10)定义了 CPU 和内存的成本

$$C^{Cost}(j) = C^{base} \times C_j \times t_{ij} + C^{Tran} \quad (9)$$

$$M^{Cost}(j) = M^{base} \times M_j \times t_{ij} + M^{Tran} \quad (10)$$

其中, C^{base} 表示资源利用率最低时的基本成本, M^{base} 表示内存空间为 1 GB 时的基本成本, t_{ij} 表示任务 T_i 在资源 R_j 运行的时间, C^{Tran}, M^{Tran} 分别表示与成本相关的 CPU 传输和内存传输.

式(11)和式(12)表示从成本函数获得的 CPU 和内存的成本模型

$$CPU(x) = \sum_{j=1}^N C^{Cost}(j) \quad (11)$$

$$Memory(x) = \sum_{j=1}^N M^{Cost}(j) \quad (12)$$

通过求解多目标优化问题可以获得最佳解决方案, 本文提出的多目标函数混合任务调度算法具体步骤由算法 1 给出.

算法 1: CSA 和 PSO 混合调度算法

步骤 1: 生成初始化种群 $2N$;

步骤 2: 随机将种群分为 2 组, 分别为 CSA 群和 PSO 群, 每组 N 个个体;

步骤 3: 目标函数 $F(x)$ 初始化, 利用式(8)评估适应度值;

步骤 4: 重复步骤 4 到 6, 直到就绪任务(K)小于最大任务数;

步骤 5: 对于每个任务(T), 使用 CSA 寻找新解;

步骤 6: 对于每个任务(T), 使用 PSO 寻找新解;

步骤 7: 利用下列公式寻找两种算法的混合解

$$z_i^d(k+1) = s \times x_i(k+1) + (1-s) \times y_i^d(k+1);$$

步骤 8: if ($z(k+1) < f(z(x))$) then

 用当前解替换 i 的解;

步骤 9: 在 $2N$ 中选择任何资源 R ;

 如果基于平台的资源 R 的执行时间较长, 则淘汰资源 R 并在 N 中另选一个资源;

步骤 10: 更新 $x_i^d(k+1) = y_i^d(k+1) = z_i^d(k+1)$ 、 p_i 和 p_g ;

步骤 11: 保留并排序最优解;

步骤 12: 结束.

3 实验结果与分析

为了评价所提算法的性能, 使用 CloudSim 3.0 仿真软件进行测试实验, 并在相同条件下与基于布谷鸟搜索和反向学习算法的任务调度方法(OCSA)^[10]、先到先得算法(FCFS)^[14]、容错蚁群优化算法(ACOw-

FT)^[15]以及基于蛾子搜索和差分进化的调度方法(MSDE)^[16]等现有任务调度方法相比. 采用完成时间、成本和最后期限违反率 3 个性能评价指标评估本文提出方法的性能. 实验中算法具体参数设置为: 种群规模 30, 惯性权重 $\omega=0.8$, 学习因子 $c_1=c_2=2$, 发现概率 $P_a=0.25$, $\alpha=1$, $\beta=0.4$.

首先, 利用任务完成时间指标评估提出调度算法的性能. 为了评估性能, 考虑 100~500 个具有不同到达率($A=10,80$)的任务. 图 1、图 2 给出了不同调度算法在到达率为 10 和 80 时的测试结果. 从图 1、图 2 中可以看出, 在相同资源下本文提出的调度算法所需的完成时间最低, 随着任务量增加, 完成时间在逐步上升. 随着到达率升高, 相同任务下的完成时间也在增加.

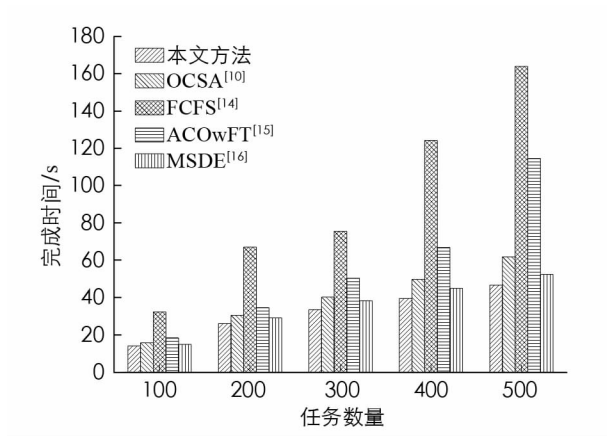


图 1 不同调度算法

在 $A=10$ 的完成时间对比

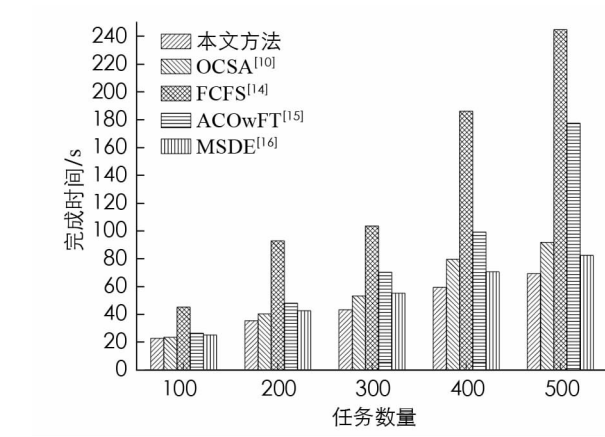


图 2 不同调度算法

在 $A=80$ 的完成时间对比

其次, 采用成本指标评估调度算法的性能. 考虑 200,500 个不同最后期限的任务, 最后期限的范围从 10~100. 图 3、图 4 给出了不同调度算法在不同任务数量的成本测试结果. 测试结果表明, 本文所提出的任务调度算法在两组任务中均达到最小代价, 而且在相同最后期限范围内随着任务量增加, 成本逐步上升.

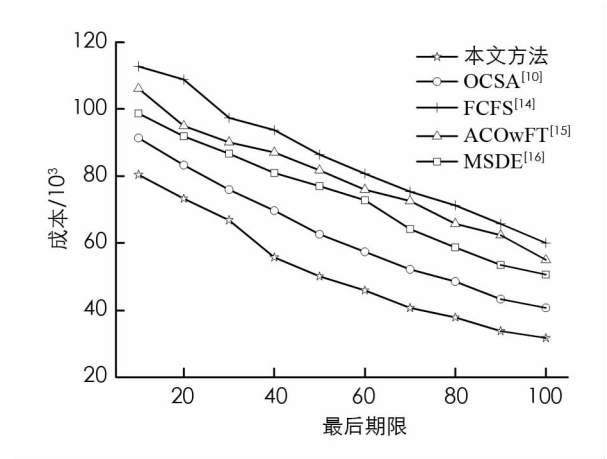


图 3 不同调度算法

在任务数为 200 时的成本对比

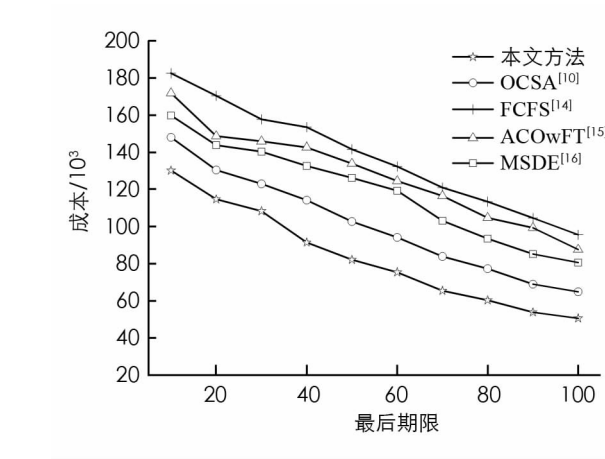


图 4 不同调度算法

在任务数为 500 时的成本对比

再次, 采用最后期限违反率指标评估本文方法的性能, 验证算法调度的服务质量(QoS). 在本次评估中, 考虑了 200 个和 500 个任务, 图 5 和图 6 显示了任务量为 200,500 时在不同最后期限情况下的违反率. 从图 5、图 6 中可以看出, 当任务数较低时, 最后期限违反率也较低, 这是因为较少的任务有更多资源可供选择. 此外, 与其他任务调度算法相比, 本文算法性能最优, 最后期限违反率最低.

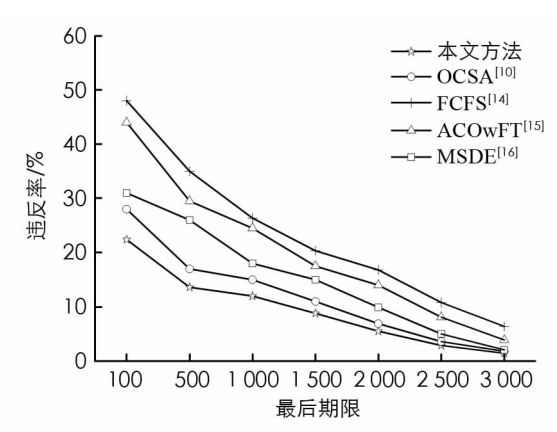


图 5 不同调度算法
在任务数为 200 时的违反率对比

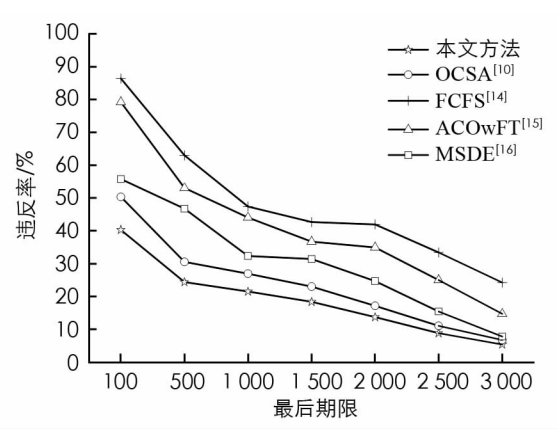


图 6 不同调度算法
在任务数为 500 时的违反率对比

4 结 语

本文提出一种基于布谷鸟搜索和粒子群优化算法的多目标优化任务调度方法, 用于解决现有云计算环境中任务调度算法在完成时间、成本和最后期限违反率等方面表现不佳的问题. 该方法通过将布谷鸟搜索和粒子群优化两种启发式智能优化算法结合在一起, 有效地提高了任务调度过程中的资源利用率, 降低了任务的完成时间、成本和最后期限违反率. 相对于其他任务调度优化算法, 本文提出的策略具有更佳的性能.

参考文献:

[1] KUMAR M, SHARMA S C. Deadline Constrained Based Dynamic Load Balancing Algorithm with Elasticity in Cloud Environment [J]. Computers & Electrical Engineering, 2018, 69: 395-411.

[2] ALMEZEINI N, HAFEZ A. Task Scheduling in Cloud Computing Using Lion Optimization Algorithm [J]. International Journal of Advanced Computer Science and Applications, 2017, 8(11): 77-83.

[3] LIN W W, WANG W Q, WU W T, et al. A Heuristic Task Scheduling Algorithm Based on Server Power Efficiency Model in Cloud Environments [J]. Sustainable Computing: Informatics and Systems, 2018, 20: 56-65.

[4] 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): 31-36.

[5] ARUNARANI A, MANJULA D, SUGUMARAN V. Task Scheduling Techniques in Cloud Computing: a Literature Survey [J]. Future Generation Computer Systems, 2019, 91: 407-415.

[6] SALEH H, NASHAAT H, SABER W, et al. IPSO Task Scheduling Algorithm for Large Scale Data in Cloud Computing Environment [J]. IEEE Access, 2019, 7: 5412-5420.

[7] AGARWAL M, SRIVASTAVA G M S. A Cuckoo Search Algorithm-Based Task Scheduling in Cloud Computing[M]// Advances in Computer and Computational Sciences. Singapore: Springer, 2017.

[8] 侯欢欢. 自适应动态调整粒子群的云计算任务调度 [J]. 计算机应用与软件, 2019, 36(9): 46-51, 116.

[9] GOBALAKRISHNAN 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.

[10] 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.

[11] DORDAIE N, NAVIMIPOUR N J. A Hybrid Particle Swarm Optimization and Hill Climbing Algorithm for Task Scheduling in the Cloud Environments [J]. ICT Express, 2018, 4(4): 199-202.

[12] LIU W L. Task Scheduling of an Improved Cuckoo Search Algorithm in Cloud Computing [J]. International Journal of Performability Engineering, 2019, 15(7): 1965-1975.

- [13] AGARWAL M, SRIVASTAVA G M S. Genetic Algorithm-Enabled Particle Swarm Optimization (PSOGA)-Based Task Scheduling in Cloud Computing Environment [J]. International Journal of Information Technology & Decision Making, 2018, 17(4): 1237-1267.
- [14] MADNI S H H, ABD LATIFF M S, ABDULLAHI M, et al. Performance Comparison of Heuristic Algorithms for Task Scheduling in IaaS Cloud Computing Environment [J]. PLoS One, 2017, 12(5): 176321-176346.
- [15] IDRIS H, EZUGWU A E, JUNAIDU S B, et al. An Improved Ant Colony Optimization Algorithm with Fault Tolerance for Job Scheduling in Grid Computing Systems [J]. PLoS One, 2017, 12(5): 177567-177590.
- [16] 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.

Multi Objective Optimization Task Scheduling Algorithm Based on CSA and PSO

WANG Li-juan¹, WU Feng²

1. School of Information Engineering, Jiaozuo Normal College, Jiaozuo Henan 454000, China;

2. School of Information Engineering, Xinyang Agriculture and Forestry University, Xinyang Henan 464006, China

Abstract: Aiming at the problems of low resource utilization, long makespan and high scheduling cost of task scheduling algorithms in existing cloud computing environments, a multi-objective optimization task scheduling strategy based on Cuckoo Search Algorithm (CSA) and Particle Swarm Optimization (PSO) has been proposed. According to the strategy, the cuckoo search algorithm and the particle swarm optimization are combined to perform the intelligent optimization task scheduling problem with the makespan, cost and deadline violation rate as the objective function, and to avoid the local optimal phenomenon in the scheduling process. The experimental results show that compared with other heuristic scheduling algorithms, the proposed method has obvious advantages, which can minimize the completion time, scheduling cost and deadline violation rate.

Key words: cloud computing; task scheduling; multi-objective optimization; cuckoo search; particle swarm optimization

责任编辑 夏 娟