

DOI:10.16644/j.cnki.cn33-1094/tp.2023.08.010

基于改进鲸鱼优化算法的资源调度策略*

余隆勇

(浙江理工大学计算机科学与技术学院, 浙江 杭州 330018)

摘要: 传统云计算在资源调度中存在效率低、计算时间分配不合理的问题;鲸鱼算法(WOA)存在收敛能力不足、容易陷入局部最优等问题。对此提出一种改进的WOA-3资源调度策略。该策略对WOA三个阶段优化,并对WOA-3和鲸鱼算法等分别进行性能优化和资源调度实验。性能优化实验表明WOA-3算法比WOA、PSO和GWO算法具有更好的收敛速度和精度;在资源调度实验中,改进的WOA-3算法相较于WOA和PSO算法分别提升了5.87%和9.76%。

关键词: 云计算;资源调度;鲸鱼算法;最优解;负载均衡;自适应权重

中图分类号:TP391

文献标识码:A

文章编号:1006-8228(2023)08-41-05

Resource scheduling strategy based on improved WOA

Yu Longyong

(School of Information Science and Technology Zhejiang Sci-Tech University, Hangzhou, Zhejiang 310018, China)

Abstract: Traditional cloud computing suffers from low efficiency and unreasonable computation time allocation in resource scheduling, and WOA suffers from insufficient convergence ability and easy to fall into local optimality. To this end, an improved WOA-3 resource scheduling strategy is proposed to optimize the three phases of WOA. Performance optimization and resource scheduling experiments are conducted for the WOA-3 and WOA, respectively. Performance optimization experiments show that WOA-3 algorithm has better convergence speed and accuracy than WOA, PSO and GWO algorithm. Resource scheduling experiments show that compared with WOA and PSO, the improved WOA-3 algorithm increases by 5.87% and 9.76%, respectively.

Key words: cloud computing; resource scheduling; whale optimization algorithm (WOA); optimal solution; load balance; adaptive weight

0 引言

群智能算法^[1]自提出以来被广泛应用于解决各种生活和科学研究问题,如资源调度、路径规划、最优策略等。它通过模拟种群中动物习性,通过单个个体的行为与群体的合作和组织,表现出群体智能行为的特征,使用随机搜索方式来获取算法最优解。但随着随着信息化的深入发展,越来越多的应用和任务部署在云端从而导致了传统群智能算法不能有效快速地解决资源分配问题。因此许多学者通过将群智能算法进行改进并运用到资源调度中,如Wu Daqin^[2]针对目

前粒子群算法性能低效,提出一种通过引入迭代选择算子的改进粒子群算法并进行实验,结果表明改进后的算法可以提高优化能力,尽可能避免陷入局部最优,具有较好的收敛效果。孟凡超^[3]等人针对云数据中心虚拟机分配存在的不均衡问题,参考用户对虚拟机资源需求,避免在同一台虚拟机上部署,提出了一种蚁群优化算法的资源调度策略,并通过实验证明该算法能够快速完成虚拟机的放置并让数据中心的虚拟机具有较低的负载不均衡度,提高资源利用率。李勋章^[4]等人针对遗传算法在资源调度中存在收敛速

收稿日期:2023-02-27

*基金项目:浙江省重点研发计划项目(2020C03094);浙江省教育厅一般科研项目(Y202147659);浙江省教育厅项目(Y202250706);浙江省基础公益研究计划项目(QY19E050003)

作者简介:余隆勇(1997-),男,江西上饶人,硕士,在读硕士,主要研究方向:云计算、分布式数据处理。

度过快以及陷入局部最优解等问题,利用多优选保留技术选取适应度值较大的个体放入种群,通过与普通个体进行杂交从而使种群往更优的方向不断进化,以此保证种群多样性以及避免陷入局部最优解的作用。程曦^[5]针对云计算资源调度优化问题,提出改进布谷鸟的随机搜索策略和丢弃概率策略,来提高算法的局部搜索能力和收敛速度,并在解决云资源调度的多目标问题针对减少延迟,提高资源利用率和服务质量取得很好的效果。

针对传统鲸鱼算法^[6]在资源调度上存在的问题,提出一种改进的鲸鱼优化算法,该算法通过对传统鲸鱼算法的三个阶段进行优化,首先通过非线性变量结构的包围阶段和基于平均最优解(Average Optimal Solution)的螺旋位置变化,提高了算法的局部优化能力和收敛速度;提出自适应加权^[7]搜索捕食策略来提高算法全局最优解的速度和精度。

1 鲸鱼优化算法

鲸鱼优化算法(Whale Optimization Algorithm, WOA)作为一种群智能算法,主要模拟座头鲸的捕食行为,并将捕食行为分为了包围猎物、螺旋气泡捕食以及搜索猎物三个阶段。

1.1 包围猎物阶段

鲸鱼在收缩包围阶段是通过向当前适应度最优的个体进行靠拢来更新位置。其余鲸鱼个体也通过这种方式来实现对猎物的包围。包围阶段位置更新公式如下:

$$X_i^{t+1} = X_i^t - A * D_{best} \quad (1)$$

$$D_{best} = |C * X_{best}^t - X_i^t| \quad (2)$$

$$A = (2r_1 - 1) * a \quad (3)$$

在式(1)、式(2)中, X_i^{t+1} 表示第 i 个个体的第 $t+1$ 次迭代, X_{best}^t 表示第 t 次迭代中最佳个体鲸鱼的位置, D_{best} 是当前 t 次迭代中的最优个体的位移, a 随着迭代次数从 2 线性减少到 0, r_1 是 (0,1) 的随机数 C 为均匀分布在 (0,2) 内的随机数。

1.2 螺旋气泡捕食阶段

螺旋气泡捕食作为鲸鱼猎食行为中进行局部搜索的另一个阶段,它通过一个螺旋式游动来对食物进行抓捕。螺旋捕食的更新公式如下:

$$X_i^{t+1} = |X_{best}^t - X_i^t| * e^{bl} * \cos(2\pi l) + X_{best}^t \quad (4)$$

其中, b 用来定义螺旋形状,通常取值为常数 1; l 是

$[-1,1]$ 之间均匀分布的随机数。包围阶段和螺旋气泡捕食阶段同作为鲸鱼算法的局部搜索阶段,在标准的鲸鱼算法中采用相同的概率 P 来决定采用哪种策略进行位置的更新。具体更新公式如下:

$$X_i^{t+1} = \begin{cases} X_{best}^t - X_i^t * e^{bl} * \cos(2\pi l) + X_{best}^t & P \leq 0.5 \\ X_{best}^t - A * D_{best} & P > 0.5 \end{cases} \quad (5)$$

1.3 搜索猎物阶段

搜索捕食是鲸鱼算法在避免陷入局部最优、求得最优解的一个阶段。在寻找最优的行为中,它主要通过随机搜索来进行鲸鱼猎食的位置更新。具体模型如下:

$$D_{best} = |C * X_{rand}^t - X_i^t| \quad (6)$$

$$X_i^{t+1} = X_{rand}^t - A |CX_{rand}^t - X_i^t| \quad (7)$$

其中, X_{rand}^t 表示当前迭代次数 t 随机生成的位置向量。搜索捕食阶段作为全局搜索过程与其他两个局部搜索过程合起来完成鲸鱼整个猎食的过程。通过 A 的值进行控制,当 $A \geq 1$ 时,算法将随机搜索一个鲸鱼个体,并通过该个体更新其他鲸鱼位置,从而使算法跳出当前猎物,寻找更加优秀的猎物。

2 改进鲸鱼优化算法

2.1 基于自适应权重的搜索捕食

传统的 WOA 算法通过 A 的取值来控制当前鲸鱼个体是否进入搜索捕食阶段,因此当 A 越大时可以使鲸鱼个体保持更佳的探索能力。或者 A 较小时采用包围策略时能够实现较高精度的求解,拥有更好的局部寻优能力。因此本文针对上述情况提出了一种基于自适应权重的方式来控制鲸鱼对策略的选取,对式(7)具体的改进如下:

$$X_i^{t+1} = X_{rand}^t - AD_{best} \quad (8)$$

$$EMBED Equation.DSMT4 = \begin{cases} \frac{1}{2} \left[1 + \sin\left(\frac{\pi}{2} + \frac{\pi t}{T_{max}}\right) \right], & t \leq \frac{T_{max}}{2} \\ \frac{1}{2} \left[1 - \sin\left(\frac{\pi}{2} + \frac{\pi t}{T_{max}}\right) \right], & t > \frac{T_{max}}{2} \end{cases} \quad (9)$$

其中, $EMBED Equation.DSMT4$ 是改进自适应搜索捕食中的控制系数,它根据当前迭代次数 t 以及改进后的正弦公式来进行下一次迭代的位置更新。

2.2 非线性变化的包围策略

在收缩和环绕阶段,通过计算每头鲸的适应度值获得当前最优解,并根据当前最优解迭代和逼近剩余的个体鲸,具体的位置更新公式如式(1)、式(2)。由

式(3)中的 a 可以得知,鲸鱼算法在进行包围行为的过程中是一种线性变化的策略,线性变化结构在算法的迭代过程容易造成趋近于局部最优解、收敛速度过快等情况。因此本文针对包围阶段提出一种非线性变化包围策略来控制算法的收敛速度。具体更新公式为:

$$X_i^{t+1} = X_{best}^t - A * |C * X_{best}^t - X_i^t| \quad (9)$$

$$a = 2 * \cos\left(\frac{\pi}{2} + \frac{t\pi}{2T_{max}}\right) \quad (10)$$

$$A = (2r_1 - 1) * a \quad (11)$$

$$C = 2 * r_2 \quad (12)$$

其中,更新后的 a 公式是一个余弦函数,它可以通过控制算法在迭代的前中后期实现非线性的变化。 r_1 、 r_2 是 0 至 1 的随机控制系数、 T_{max} 是最大迭代次数。在更新后的公式中,鲸鱼收敛速度随着迭代次数的增加是越来越快的,但是在后期会减缓来保证算法的收敛能力从而更好的求最优解、减少收敛时间。

2.3 基于 AOS 的螺旋位置变更

鲸鱼的捕食路径是一种螺旋状,鲸鱼个体可能会向最优个体靠拢来进行位置的更新,这种单一的更新方式很容易导致陷入局部最优解问题,当其余鲸鱼个体也向当次迭代最优解靠拢时,就容易导致所有个体产生局部最优解。在该阶段本文通过借鉴灰狼算法^[9]思想,提出基于 AOS(Average Optimal Solution)的螺旋位置变更策略。具体的公式为:

$$X_{new}^t = \omega_1 * X_{best1}^t + \omega_2 * X_{best2}^t + \omega_3 * X_{best3}^t \quad (13)$$

$$\omega_1 + \omega_2 + \omega_3 = 1 \quad (14)$$

$$X_i^{t+1} = \begin{cases} X_{new}^t - X_i^t * e^{bl} * \cos(2\pi l) + X_{best}^t & P \leq 0.5 \\ X_{new}^t - \varphi * A * D_{best} & P > 0.5 \end{cases} \quad (15)$$

其中, ω_1 、 ω_2 、 ω_3 为[0,1]的随机数且总和为 1, X_{best1}^t 、 X_{best2}^t 、 X_{best3}^t 为当前种群中最优的三个鲸鱼个体。通过不同的变更系数 ω 与其对应的最优个体向量的成积和来进行鲸鱼个体的螺旋位置更新。该算法采用三个最优个体的组合作为更新策略,保证了求解过程中收敛速度不会过快,增加了鲸鱼种群的多样性,避免了基于单个最优个体的位置更新陷入局部最优解的问题。改进位置公式如下:

$$X_i^{t+1} = |X_{new}^t - X_i^t| * e^{bl} * \cos(2\pi l) + X_{best}^t \quad (16)$$

2.4 WOA-3 改进算法具体流程设计

改进后的 WOA-3 算法在搜索捕食阶段中,将一个鲸鱼个体当作一个具体的解,根据每一次迭代的

结果对鲸鱼个体进行评估,在达到最大迭代次数时将在当前最大迭代次数中的最优鲸鱼个体视为算法的全局最优解。具体步骤如下。

(1) 初始化鲸鱼种群:种群规模大小 n ,为每个鲸鱼个体随机生成初始位置,螺旋位置更新系数 ω_1 、 ω_2 、 ω_3 , 最大迭代次数 T_{max} ,参数 a 、 A 、 C 、 r_1 、 r_2 、 P 等。

(2) 计算种群初始化适应度,找到当前最优解 X_{best}^t 表示当前迭代最优鲸鱼个体的位置。

(3) 进入迭代,更新参数参数 a 、 A 、 C 、 r_1 、 r_2 、 P 。根据当前迭代结果计算每个鲸鱼的适应度值并进行排序,得出当前迭代中前三个最优解 X_{best1}^t 、 X_{best2}^t 、 X_{best3}^t 以及 X_{new}^t 。

(4) 当 $P > 0.5$ 且 $A > 1$ 时,选择公式(10),采用随机一个 X_{rand}^t 用来进行鲸鱼的位置更新;当 $P \leq 0.5$ 且 $A > 1$ 时,算法通过公式(15)进行鲸鱼个体的位置更新。

(5) 当 $P > 0.5$ 通过公式(16)确定鲸鱼个体的位置更新。

(6) 根据 2-5 过程更新当前迭代的种群适应度值并重新排序,更新当前解 X_{best1}^t 、 X_{best2}^t 、 X_{best3}^t 以及 X_{new}^t 。

(7) 如果当前个体的最新解优于上次迭代的值,则进行更新。

(8) 如果当前迭代次数达到最大值 T_{max} ,输出当前鲸鱼最优解;未达到最大迭代次数,跳回步骤(2)。

3 仿真实验与结果

3.1 改进鲸鱼优化算法性能实验

为了验证本文提出的改进鲸鱼优化算法^[10]性能,我们将选取 WOA、POS、GSA 和本文改进的 WOA-3 对四个测试函数在 Matlab 进行实验,具体函数如表 1 所示。得到对应的最优解、平均解和标准差,如表 2 所示。

表 1 测试函数

函数名	函数公式
F1	$\sum_{i=1}^d x_i^2$
F2	$\sum_{i=1}^d x_i + \prod_{i=1}^d x_i $
F3	$\sum_{i=1}^d -x_i \sin\left(\sqrt{ x_i }\right)$
F4	$\sum_{i=1}^n (x_i + 0.5)^2$

由表 2 可知,标准 WOA 算法在四个函数测试的最优解、平均值以及标准差在大多数情况下都是优于 PSO 和 GWO 算法的,但是改进的 WOA-3 算法在这些结果中相比于 WOA 算法,除了在 f4 函数的结果存在略微偏差,其余函数的计算精度均有着明显的提高。

为了能够表明改进算法的作用,本次实验还将各算法在这四个测试函数中的收敛情况进行对比,结果显示,WOA算法比PSO和GWO算法有更好的收敛性,改进的WOA-3算法比WOA算法有更好的收敛速度和求解精度,具体如图1~图4所示。

表2 测试函数结果

函数	指标	WOA	PSO	GWO	WOA-3
F1	最优值	4.40E-7	7.21E-0	1.01E-3	3.09E-3
	平均值	3.92E-6	3.36E-0	1.34E-2	4.61E-3
	标准差	1.32E-6	8.81E-0	2.80E-2	0.00E-0
F2	最优值	1.43E-5	6.45E-0	2.05E-1	1.73E-1
	平均值	1.05E-4	1.74E-0	8.86E-1	2.29E-1
	标准差	2.65E-4	2.50E-0	5.49E-1	1.09E-1
F3	最优值	3.29E-0	6.72E-4	6.61E-2	2.44E-0
	平均值	1.05E-0	9.20E-2	5.25E-0	1.35E-0
	标准差	5.53E-0	7.21E-1	2.74E-2	8.57E-2
F4	最优值	2.77E+0	2.73E+0	2.58E+2	2.62E+0
	平均值	3.39E-0	1.30E+3	2.74E+1	2.71E+0
	标准差	2.85E+0	7.75E+9	8.72E-0	5.28E-0

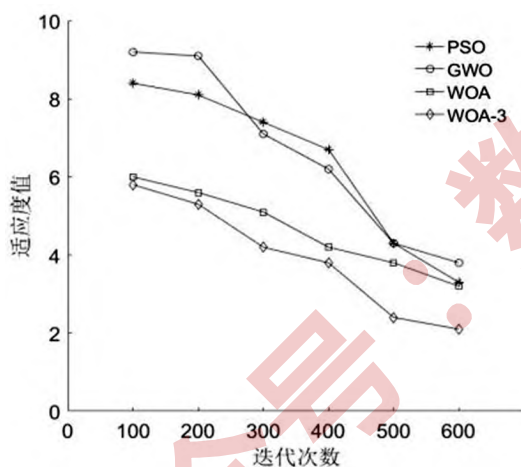


图1 F1函数

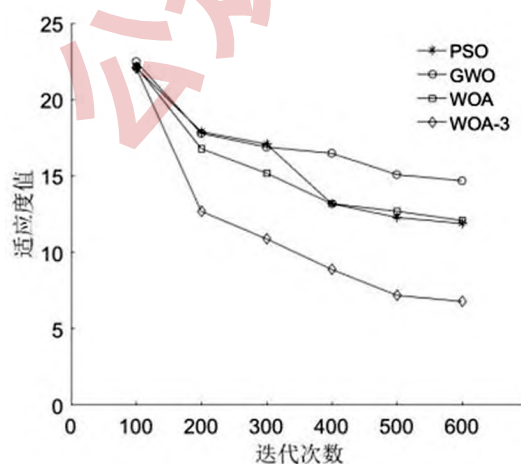


图2 F2函数

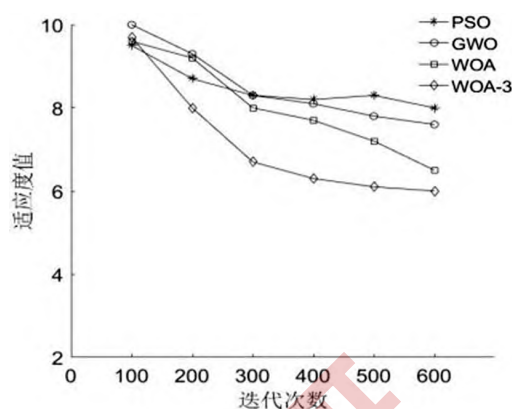


图3 F3函数

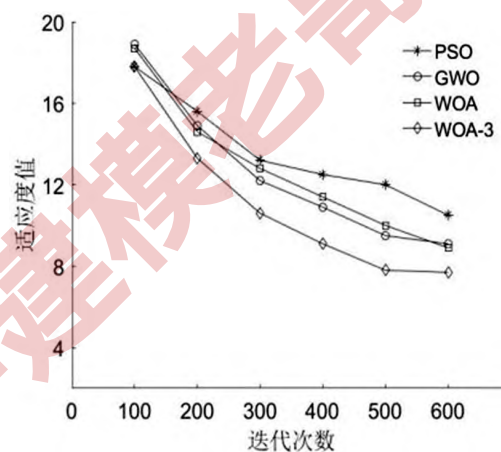


图4 F4函数

3.2 资源调度仿真实验验证

3.2.1 实验环境

为了验证改进的WOA-3算法在资源调度中的性能,本文通过CloudSim^[8]平台模拟在Kubernetes集群上进行资源调度的实验。集群主要包括一个主节点Master和四个Node节点,并使用十个Pod任务进行测试。Master节点用于实现资源调度、监控等功能,而Node节点则是真正用来处理Pod任务请求的工作节点。具体配置如表3和表4所示。

表3 实验配置信息

节点名	CPU/核	内存/G	磁盘大小/GB	网络带宽MB/s
Node1	2	4	20	20
Node2	2	4	20	20
Node3	2	4	20	20
Node4	2	4	20	20

3.2.2 实验结果与分析

在本文中,为了验证WOA-3算法在资源调度上的性能将其与WOA、PSO以及Kubernetes的默认调度算法在资源调度性能进行对比。相比于Kubernetes的

默认性能评价因子CPU和内存,在实际的应用场景中任务的调度还需要考虑到对网络宽带的访问和磁盘的读写需求,所以在本次实验中额外添加了磁盘和网络两个性能指标。从图5~图8可以看出,相比于PSO和默认资源调度算法相比,WOA算法得到了改进,但改进后的WOA-3算法可以使资源负载更加均衡,资源利用率高于WOA算法。由此说明,改进的WOA-3算法在进行资源调度时能够让各节点的利用率更好,使负载更加均衡。

表4 Pod请求资源量

Pod	CPU/核	内存/MB	磁盘/MB	网络/MB
Pod1	0.2	200	10	5
Pod2	0.2	300	10	10
Pod3	0.2	400	10	15
Pod4	0.3	200	20	5
Pod5	0.3	200	20	10
Pod6	0.3	400	20	15
Pod7	0.4	200	30	5
Pod8	0.4	300	30	10
Pod9	0.4	400	30	15

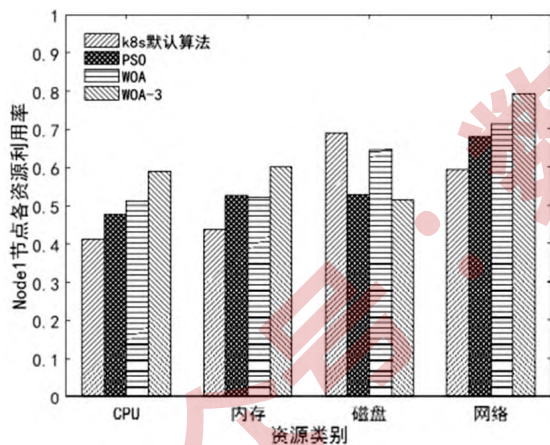


图5 Node1节点

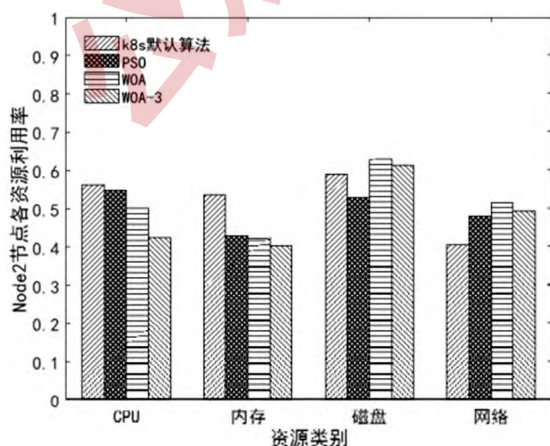


图6 Node2节点

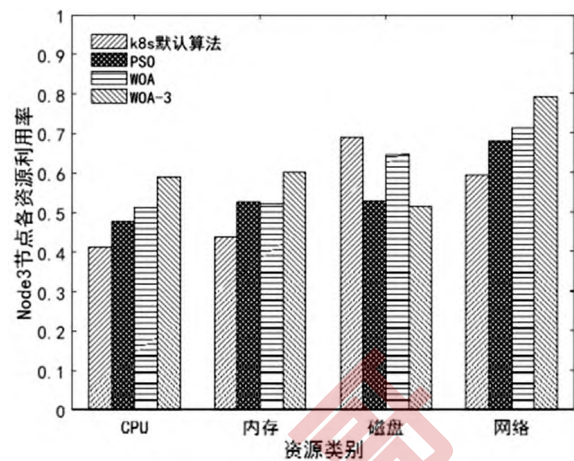


图7 Node3节点

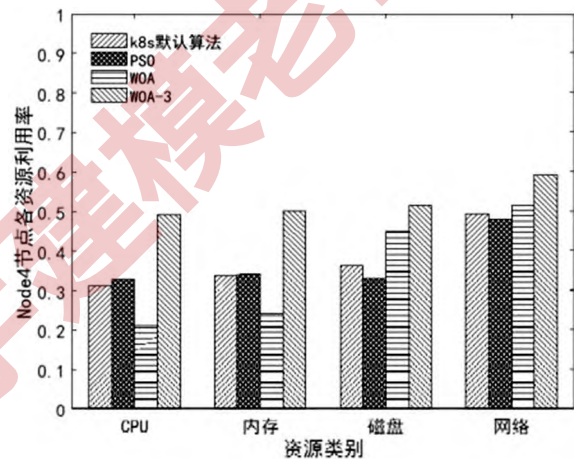


图8 Node4节点

4 结束语

针对鲸鱼算法在资源调度中容易出现收敛速度慢,陷入局部最优解等问题,提出一种改进的WOA-3算法。WOA-3算法主要是针对鲸鱼猎食三个阶段分别进行改进,通过改进非线性变化结构的包围策略以及基于AOS策略改变螺旋更变位置,来加快鲸鱼算法的局部收敛速度;同时,通过提出自适应加权捕食策略来提高鲸鱼算法的全局寻优能力和收敛能力。在算法性能仿真实验中验证了算法的收敛性和优化能力,改进的WOA-3算法相比于传统的WOA、PSO和GWO算法,都有更好的收敛性,能够在更少的迭代中趋近于最优值;同时也在资源调度的仿真实验中验证了WOA算法相比于Kubernetes默认的调度算法和PSO算法有一定的提升,而改进WOA-3算法相比于传统WOA算法能够使资源的调度更加均衡,资源利用率更高。

(下转第51页)

本文主要研究了自然场景的中英文文本识别,而对于提高模型准确率、识别速度以及硬件移植将成为未来研究的重点。

参考文献(References):

- [1] Liu C, Cao Y, Luo Y, et al. Deepfood: Deep learning-based food image recognition for computer-aided dietary assessment[C]//International Conference on Smart Homes and Health Telematics. Springer, Cham, 2016:37-48
- [2] Kahn G, Villafior A, Ding B, et al. Self-supervised deep reinforcement learning with generalized computation graphs for robot navigation[C]//2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018:5129-5136
- [3] Shen Z, Zhang R, Dell M, et al. LayoutParser: A unified toolkit for deep learning based document image analysis[C]//International Conference on Document Analysis and Recognition. Springer, Cham, 2021:131-146
- [4] 卢官明. 基于内容的图像及视频检索[J]. 南京邮电大学学报: 自然科学版, 2002, 22(2): 23-26
- [5] Shi B, Bai X, Yao C. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition[J]. IEEE transactions on pattern analysis and machine intelligence, 2016, 39(11): 2298-2304
- [6] Lee C Y, Osindero S. Recursive recurrent nets with attention modeling for ocr in the wild[C]//Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 2231-2239
- [7] Sheng F, Chen Z, Xu B. NRTR: A no-recurrence sequence-to-sequence model for scene text recognition[C]//2019 International conference on document analysis and recognition (ICDAR). IEEE, 2019: 781-786
- [8] Qiao Z, Zhou Y, Yang D, et al. Seed: Semantics enhanced encoder-decoder framework for scene text recognition[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 13528-13537
- [9] Graves A, Fernández S, Gomez F, et al. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks[C]//Proceedings of the 23rd international conference on Machine learning, 2006: 369-376
- [10] 孙洁, 朱玉全, 黄承宇. 基于二维 CTC 和注意力序列的场景文本识别模型[J]. 电子制作, 2022, 30(17): 65-70
- [11] Shi B, Wang X, Lyu P, et al. Robust scene text recognition with automatic rectification[C]//Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 4168-4176
- [12] Cheng Z, Xu Y, Bai F, et al. Aon: Towards arbitrarily-oriented text recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: 5571-5579
- [13] Luo C, Jin L, Sun Z. Moran: A multi-object rectified attention network for scene text recognition[C]//Pattern Recognition, 2019, 90: 109-118
- [14] Yang M, Guan Y, Liao M, et al. Symmetry-constrained rectification network for scene text recognition[C]//Proceedings of the IEEE/CVF international conference on computer vision, 2019: 9147-9156

(上接第 45 页)

参考文献(References):

- [1] 袁浩, 李昌兵. 基于社会力群智能优化算法的云计算资源调度[J]. 计算机科学, 2015, 42(4): 206-208, 243
- [2] Wu D. Cloud computing task scheduling policy based on improved particle swarm optimization[C]//2018 International Conference on Virtual Reality and Intelligent Systems (ICVRIS). IEEE, 2018: 99-101
- [3] 孟凡超, 张海洲, 初佃辉. 基于蚁群优化算法的云计算资源负载均衡研究[J]. 华中科技大学学报(自然科学版), 2013, 41(S2): 57-62
- [4] 李勋章, 王如月, 莫静容. 改进的遗传算法在云资源调度上的应用[J]. 桂林航天工业学院学报, 2022, 27(1): 9-13
- [5] 程曦, 宋铁成. 基于改进多目标布谷鸟搜索的资源调度算法[J]. 计算机应用与软件, 2022, 39(3): 241-246, 253
- [6] 许德刚, 王再庆, 郭奕欣, 等. 鲸鱼优化算法研究综述[J]. 计算机应用研究: 1-10[2022-12-05]
- [7] 文斌, 朱瞻. 基于自适应权重的立体匹配优化算法[J]. 计算机工程, 2021, 47(4): 268-276
- [8] 张勋, 张呈宇, 魏进武. 一种基于容器编排技术的资源运维系统的设计与实现[J]. 电信科学, 2018, 34(6): 162-170
- [9] 孟荣华, 孙艾文, 吴正佳, 等. 改进灰狼算法求解复杂混合流水调度问题研究[J]. 电子学报, 2021, 49(8): 1515-1523
- [10] 王延峰, 廖荣航, 梁恩豪, 等. 基于围攻机制的改进鲸鱼优化算法[J]. 控制与决策: 1-9[2022-12-01]