

基于人工鱼群算法的容器云资源低能耗部署方法

徐胜超, 杨波

(广州华商学院 数据科学学院, 广东 广州 511300)

摘要:目前容器云资源部署过程中能耗较高,直接增加了云服务提供商的经济成本。提出基于人工鱼群算法的容器云资源低能耗部署方法。首先,对容器平台的能量消耗模型展开详细分析,确定容器相关运行参数;通过制定的模型约束条件建立容器云资源的低能耗部署模型;然后,使用人工鱼群算法对模型求解,搜索出部署模型的全局最佳值,制定最佳部署方案;最后,依据制定的资源低能耗部署方案,实现容器云资源的低能耗部署。测试容器云资源低能耗部署时的平均最大等待时间、最大响应时间、最大平均任务队列长度和平均能量损耗。测试结果表明,容器云资源低能耗部署时的平均最大等待时间在检测次数为50次时,仍未超过70 ms;最大响应时间在检测次数为50次时未超过45 ms;随着部署时间增加,最大平均任务队列长度为85 mm,10次实验中平均能量损耗均在25 672 kJ以内,由此可见该方法在容器云资源低能耗部署中具有较好的性能。

关键词:人工鱼群算法;容器;云资源;低能耗;部署方法

中图分类号:TP391

文献标识码:A

文章编号:1673-629X(2023)06-0022-06

doi:10.3969/j.issn.1673-629X.2023.06.004

Low Energy Consumption Deployment Method for Container Cloud Resources Based on Artificial Fish Swarm Algorithm

XU Sheng-chao, YANG Bo

(School of Date Science, Guangzhou Huashang College, Guangzhou 511300, China)

Abstract: High energy consumption during the deployment of container cloud resources has increased the economic cost of cloud service providers. A low energy consumption deployment method of container cloud resources based on artificial fish swarm algorithm is proposed. Firstly, the energy consumption model of the container platform is analyzed in detail, and the relevant operation parameters of the container are determined. The low energy consumption deployment model of container cloud resources is established through the established model constraints. Then, the artificial fish swarm algorithm is used to solve the model, search the global optimal value of the deployment model, and formulate the optimal deployment scheme. Finally, the low energy consumption deployment of container cloud resources is realized according to the developed resource low energy consumption deployment scheme. The average maximum waiting time, maximum response time, maximum average task queue length, and average energy consumption in the low energy consumption deployment of the container cloud resources are tested. It is showed that the average maximum waiting time for low energy consumption deployment of container cloud resources does not exceed 70 ms when the number of inspections is 50. The maximum response time does not exceed 45 ms when the number of tests is 50. As the deployment time increases, the maximum average task queue length is 85 mm, and the average energy loss in 10 experiments is within 25 672 kJ. Therefore, the proposed method has excellent performance in the low energy consumption deployment of container cloud resources.

Key words: artificial fish swarm algorithm; container; cloud resource; low energy consumption; deployment method

0 引言

如何让云计算变得更加节能高效,是当今政府和企业都在关注的重点问题^[1-2]。容器技术^[3]是近年来云计算行业中重要的一部分,容器虚拟化技术及其平台更是凭借自身部署快、性能高、启动迅速等优

势,被广泛应用于各大云服务项目中。但是容器云资源长期运行时,会增加不必要的能耗消耗,为提升容器的使用性能,降低容器云资源能耗,提出高效的容器云资源低功耗部署方法^[4],就变得尤为迫切。

文献[5]提出基于多维资源空闲率权重的容器云

收稿日期:2022-07-27

修回日期:2022-11-29

基金项目:国家自然科学基金项目(61772221);广州华商学院校内导师制科研项目资助(2022HS07)

作者简介:徐胜超(1980-),男,硕士,讲师,研究方向为并行分布式处理软件。

多维资源利用率均衡调度方法,该方法综合考虑物理节点 CPU、内存、磁盘、网络带宽空闲率和已部署的容器应用个数等因素,利用模糊层次分析法自动建模求解容器应用多维资源权重参数。文献[6]提出基于遗传算法的容器云资源配置优化方法,该方法将容器云平台数据中心整体能耗最低作为目标函数,设置物理主机与虚拟机对应、虚拟机与容器对应等约束条件,利用遗传算法通过染色体表达、初始化、交叉操作、变异操作以及设置适应度函数 5 个步骤求解目标函数,获取最优容器云环境资源配置结果。文献[7]提出容器云环境虚拟资源配置策略的优化方法,首先,提出容器云虚拟资源的配置和迁移方案,发现物理机选择策略对数据中心能耗有重要影响;其次,通过研究主机利用率与容器利用率,主机利用率与虚拟机利用率,主机利用率与数据中心能耗之间的数学关系,建立容器云数据中心能耗的数学模型,定义出优化目标函数;最后,通过对物理机的能耗函数使用线性插值进行模拟,依据邻近事物相类似的特性,提出改进的最佳能耗优先物理机选择算法。上述三种方法在提高容器云资源利用效率方面都有很好的改善,但是应用到云资源部署的过程中还是存在能耗较高的问题,影响了容器资源部署效果。

人工鱼群算法是通过模仿鱼类行为方式,从而提出出来的一种基于动物自治体的优化方法,它能够很好地解决非线性函数优化等问题。人工鱼群算法的收敛速度较快,可以用来解决有实时性要求的问题,针对一些精度要求不高的情况,可以用来快速得到一个可行解。由于容器在云资源低能耗部署过程中,需要通过用户需求调整容器低能耗部署方案,是动态优化问题,对于模型的求解效率要求较高。因此,该文通过人工鱼群算法快速确定云资源能耗极值变化时间,寻找出满意的解域,制定最佳的云资源低能耗部署方案。

1 低能耗部署模型的建立

1.1 容器云及其能量消耗模型

容器^[8-9]是提供程序运行的独立空间, Docker 容器平台负责容器的生命周期管理以及相关资源信息的管理。 Docker 容器技术^[10]的出现,为容器平台的诞生打下了坚实的基础。

容器云平台开展资源低功耗部署时,平台基本单位为容器,封装软件为平台运行环境。容器云平台主要侧重于容器的编排以及资源的部署、共享,具体的资源组织结构如图 1 所示。

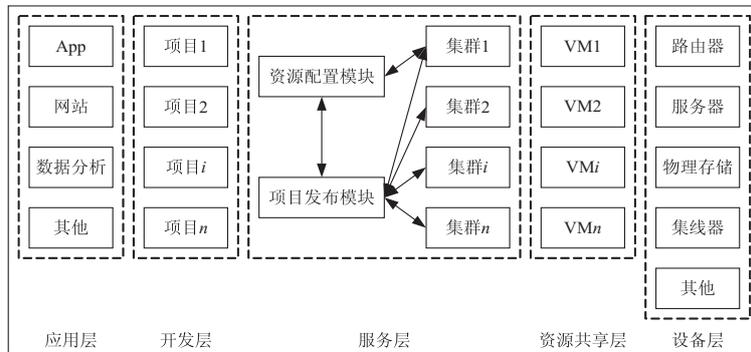


图 1 容器云平台资源组织结构

容器云平台^[11-12]包括应用、开发、资源共享、设备、集群服务等层级。资源配置模块与项目开发模块位于集群服务层中,该文将研究的云资源低能耗部署方法应用于该资源配置模块中,实现云资源的低能耗部署配置。

设有 m 个物理主机和 n 个容器的云平台中,云资源部署的能量消耗主要来自固定和迁移两部分,经实验发现,云资源能耗与物理主机的资源利用率有较大关联,因此 t 时刻物理主机的能耗见公式(1):

$$E_{\text{host}}(t) = E(U_{\text{cpu}}(t)) + E(U_{\text{mem}}(t)) + E(U_{\text{disk}}(t)) + E(U_{\text{bw}}(t)) \quad (1)$$

其中, $E(U_{\text{cpu}}(t))$ 表示物理主机 CPU 的能耗, $E(U_{\text{mem}}(t))$ 表示物理主机的内存能耗, $E(U_{\text{disk}}(t))$ 表示物理主机的磁盘能耗, $E(U_{\text{bw}}(t))$ 表示物理主机的

网络带宽能耗。这样第 i 个物理主机在 $[t_a, t_b]$ 时间段的总体能量消耗 E_i 可以按照公式(2)来计算:

$$E_i = \int_{t_a}^{t_b} E_{\text{host}}(t) dt \quad (2)$$

考虑到容器云资源调度过程中的能耗除了硬件资源利用率实时变化产生的能耗外,还包括容器迁移能耗。基于系统各部位的资源利用率以及能耗公式得出容器迁移开销,其计算见公式(3):

$$V_j = \int_{t_1}^{t_1+t_{kj}} v_j(t) dt \quad (3)$$

其中, j 代表容器, $v_j(t)$ 代表 t 时刻的第 j 个容器处理器利用率, t_1 代表容器开始迁移的时间, t_{kj} 代表容器完成全部迁移所需的时间, V_j 代表容器迁移开销。

由于容器迁移开销是影响容器云资源调度能耗的重要因素,因此容器云资源能耗分析过程中重点

计算容器迁移开销。整个容器云平台的能量消耗见公式(4):

$$E_{\text{total}} = \sum_{i=1}^m E_i + \sum_{j=1}^n V_j \quad (4)$$

1.2 容器迁移

由于云计算环境中的任务呈现多样性,导致任务在处理时不仅有实时的在线处理任务,还有若干异步任务。所以云数据中心运行时,容器、虚拟机^[13-14]以及物理机都会随着任务的进行增加负载情况。其中,由于物理机任务完成时间较早,所以当物理机完成任务,而容器未完成任务时,会出现短暂的资源浪费。

在容器云环境中,虚拟机迁移是保障物理服务器负载均衡的关键。迁移过程中,通常分为触发迁移时间、迁移目标服务器以及迁移位置 3 个阶段。迁移触发时间主要依据数据中心的资源部署策略来实现。迁移过程如下:

(1)统计物理主机资源利用率,通过统计将物理主机^[15]中过载的利用率信息保存在相关列表中。

(2)基于列表信息,对其展开降序排列,选取对应的容器迁移目标。

(3)依据排列的顺序从列表选取对应物理主机,并将其作为迁移过程中的源物理主机,依据顺序完成迁移后,保证源物理主机利用率不超载,若超载则迁移不成功,需要依据上述流程继续寻找迁移目标。

(4)统计物理主机中资源利用率潜在信息,并将其存储在相应列表中。

(5)选取一台不在列表当中的物理主机作为容器迁移目标,将部署于源物理主机上的容器全部迁移至目标物理主机中,完成容器的迁移。

依据上述容器云虚拟资源的迁移流程可知,容器在云资源部署以及迁移过程中,容器云都需要面对物理主机的选取问题,物理主机是整个数据中心^[16]的最大能源消耗部件,所以,找出物理主机与容器、虚拟机之间的能耗计算关系,是完成容器云资源低能耗部署的关键。

1.3 约束条件设置

容器运行时,可以通过 Hypervisor 采集容器的虚拟资源运行状态参数,但是由于采样参数过多,会对物理机的运算带来额外能耗负担,所以该文在建立数据中心能耗模型时,通过灰色关联分析方法^[17],完成容器运行状态参数的提取。设定容器虚拟能耗为 P ,容器运行参数比较序列设定为 (p_1, p_2, \dots, p_n) ,通过关联系数计算方法,完成参数比较序列的采样,过程如下式所示:

$$\beta_i(P, a_i) = \frac{1}{n} \sum_{l=1}^n \beta(P(l), p_i(l)) \quad (5)$$

式中,容器比较序列与 P 之间的灰色关联值标记 $\beta(P, p_i)$ 形式,容器序列的两级最小值分别用 $P(l)$ 、 $p_i(l)$ 表述,常数标记 i 、 n 形式。设定 $\beta_i \geq 0.6$ 为参数选取阈值,完成容器运行状态的选取^[18-19]。

2 基于人工鱼群算法的容器云资源低能耗部署方法

设定人工鱼群个体状态值为 Q_i ,食物浓度标记 P_i ,鱼群视野范围标记 σ ,最大移动步长标记 u ,通过人工鱼群的行为分析,总结鱼群觅食行为,获取容器云资源能耗部署模型全局最佳值,完成最佳部署方案的制定。流程如下:

步骤 1:初始化人工鱼群参数,建立人工鱼群,并通过容器云资源部署模型对鱼群实施编码处理,预测任务执行时间。

步骤 2:开展鱼群个体觅食、聚群追尾等行为,估计当前状态下,任务搜索所需时间。

步骤 3:基于评价函数完成鱼群行为评价,剔除个别个体,更新鱼群行为。过程中,若迭代次数大于设定的阈值,则需要选取相应种群个体完成跳跃行为。

步骤 4:若鱼群多次跳跃后公告板差值不变,鱼群结束搜索,将搜索结果中的最佳全局值,算法结束。

鱼群编码^[20-21]就是将容器云资源低能耗部署问题的解空间与算法搜索空间互相映射的过程。根据容器云虚拟资源的迁移过程,使用人工鱼群算法对容器云资源的低能耗部署模型进行求解。设定容器数量为 x ,物理主机数量为 y ,二进制的编码搜索空间设定 2^{xy} ,多值编码搜索空间为 y^x ,从而完成鱼群编码 $b = (c_1, c_2, \dots, c_x)$ 。

建立评价函数:

人工鱼群搜索过程中,需要建立相应的评价函数确定鱼群个体的优劣,容器云资源能耗部署模型具备相关约束条件,所以若鱼群寻找的最佳值不能满足约束条件,需要建立相关的目标函数,辅助鱼群完成全局最佳值的搜索,建立的容器云资源低能耗部署模型的目标函数如下式所示:

$$f(x) = \begin{cases} \eta_1 \cdot D + \eta_2 \cdot E + \eta_2 \cdot F, \text{满足约束} \\ \eta_1 \cdot D + \eta_2 \cdot E + \eta_2 \cdot F + \delta \cdot G, \text{不满足约束} \end{cases} \quad (6)$$

式中,建立的容器云资源低能耗部署模型目标函数标记 $f(x)$ 形式,惩罚系数标记 δ 形式,目标向量用 η 表述,容器云资源低能耗部署模型相关参数标记 D 、 E 、 F 、 G 形式。

基于上述建立的容器云资源低能耗部署模型目标

函数,结合容器资源部署时的负载均衡问题,建立评价函数,过程如下式所示:

$$g(x) = 1/f(x) \quad (7)$$

式中,建立的鱼群个体评价函数用 $g(x)$ 表述。

人工鱼群中的个体数量越多,收敛速度越快,搜索精度越高。依据弱肉强食原则,在人工鱼群算法运行时,剔除搜索性能较差的个体,这样不仅能够满足算法初期对收敛速度的要求,还能够降低算法的复杂程度。搜索性能较差的个体离部署模型的最优解较远,对鱼群的影响较为负面,剔除该人工鱼有利于算法的收敛。

跳跃行为有利于提升人工鱼群算法的全局搜索能力。鱼群出现跳跃行为,说明云资源集群中节点差异较小,鱼群容易进入局部最优。跳跃行为能够强行改变鱼群的参数,使鱼群跳出局部区域,保障鱼群的全局搜索能力。

通过上述人工鱼群算法^[22-23],调整容器云资源低能耗部署策略,建立容器云资源低能耗最佳部署方案,就实现了容器云资源的低能耗部署。

3 仿真实验与性能分析

3.1 实验设置

为了验证上述容器云资源低能耗部署方法的整体有效性,需要对此方法展开测试。实验环境配置如表1所示。

表1 实验环境配置

操作系统	Win 10
CPU	Inter(R) Core(TM) i5-2450M
内存	4 GB
硬盘	500 G
软件包	CloudSim5.0
虚拟机个数/个	800
物理主机台数/台	60

3.2 参数设置与比较对象

3.2.1 参数设置

设置实验初始条件如表2所示。

表2 实验初始参数设置

实验初始条件	数量设置
人工鱼数量/个	50
最大循环次数/次	100
节点总数/个	60
最大备份任务数/个	2
节点快慢阈值	0.8
任务执行速度/ms	0.3

3.2.2 比较对象

实验选取云资源部署平均最大等待时间、最大响

应时间、容器最大平均任务队列长度以及容器云资源能量损耗4个指标对容器资源部署方法进行测试。分别采用基于人工鱼群算法的容器云资源低能耗部署方法(方法1)、文献[5]方法(方法2)、文献[6]方法(方法3)、文献[7]方法(方法4)检测4种方法的资源部署效果。

3.3 测试结果分析

(1)云资源部署平均最大等待时间测试。

云资源部署平均最大等待时间是容器云资源低能耗部署时的最长平均延迟时间,数值越高部署效果越差。采用方法1、方法2、方法3以及方法4开展容器资源低能耗部署时,测试上述4种方法的平均最大等待时间,测试结果如图2所示。

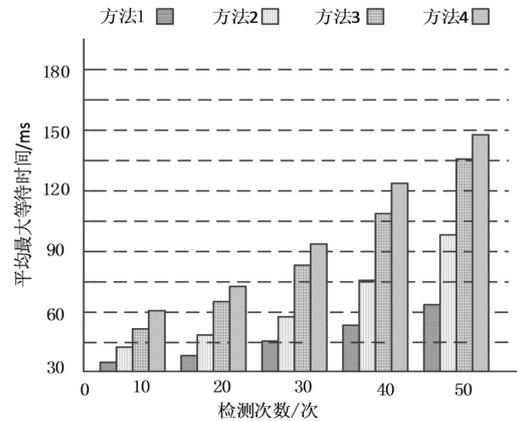


图2 不同方法的平均最大等待时间测试结果

分析图2实验数据可知,方法2、3、4测试出的平均最大等待时间在第10次时均超过40ms,最高为60ms,而方法1平均最大等待时间仅33ms;随着检测次数的增加,4种方法测试出的平均最大等待时间都呈现出不同程度的上升趋势。但同次数测试结果对比,方法1测试出的容器云资源部署平均最大等待时间是4种方法中最低的,检测次数为50时平均最大等待时间不超过70ms,由此可见应用方法1可缩短平均最大等待时间。这主要是因为方法1在容器云资源低能耗部署前对容器云资源配置开展了详细的分析,并且通过人工鱼群算法,调整容器云资源低能耗部署策略,建立容器云资源低能耗最佳部署方案,所以该方法在云资源低能耗部署时的平均最大等待时间短。

(2)最大响应时间测试。

最大响应时间是容器云资源部署过程中的最长延迟时间。采用方法1、方法2、方法3以及方法4开展容器资源低能耗部署时,测试上述4种方法的最大响应时间,测试结果如图3所示。

测试过程中,部署最大响应时间越长,说明部署过程的部署效果越差,反之则越好。分析图3可知,方法1、2、3、4在检测次数为10次时,最大响应时间相差

大,均低于 30 ms,方法 1 最大响应时间最短;随着检测次数的增加,4 种方法测试出的最大响应时间都出现了不同程度的上升,方法 3 上升效果最明显。方法 1 的最大响应时间均未超过 45 ms,其他方法最大响应时间最长超过 100 ms;对比同次数测试结果,方法 1 测试出的容器云资源低能耗部署响应时间是 4 种方法中最低的,说明方法 1 在开展容器资源低能耗部署时的部署效果最优。是因为该文通过制定的模型约束条件建立容器云资源的低能耗部署模型,再使用人工鱼群算法对模型求解,搜索出部署模型的全局最佳值,制定最佳部署方案,从而提升了部署效率。

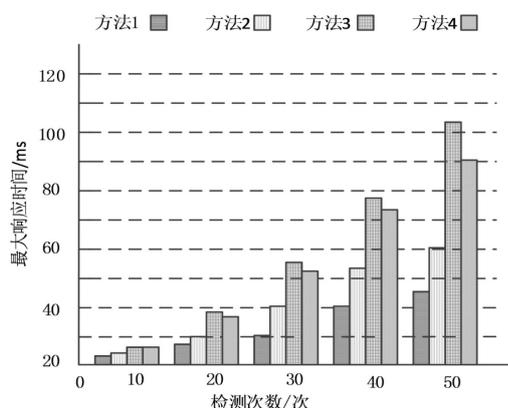


图 3 不同方法的最大响应时间测试结果

(3) 容器最大平均任务队列长度测试。

容器最大平均任务队列长度能直观地体现容器资源部署时的容器负载情况。采用方法 1、方法 2、方法 3 以及方法 4 开展容器资源低能耗部署时,测试上述 4 种方法的容器最大平均任务队列长度,查看 4 种方法的容器负载情况。测试结果如图 4 所示。

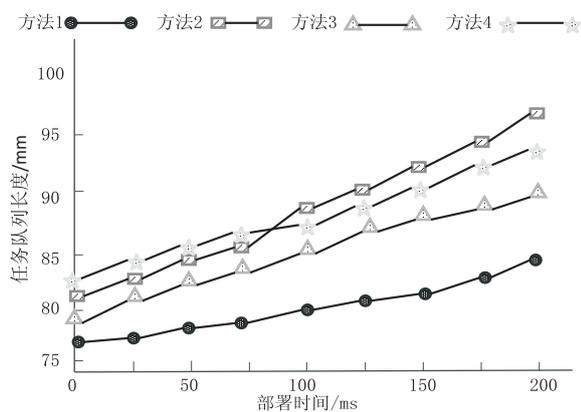


图 4 最大平均任务队列长度测试结果

最大平均队列长度是容器运行时间内,容器部署任务队列的平均长度最大值,实验过程中,任务队列长度越长,说明部署方法的排队时间越长,容器负载就越大,反之则越小。分析图 4 可知,方法 1 在初始部署任务时,与方法 2、3、4 对比队列长度最短。随部署时间增大,4 种方法的任务队列长度均有所增长;但比较 4

种方法,方法 1 在相同的部署时间内测试出的任务队列长度是 4 种方法中最短的,说明方法 1 在云资源部署时的容器负载小。是因为该文基于建立的容器云资源低能耗部署模型的目标函数,结合容器资源部署时的负载均衡问题,建立评价函数,并利用全局搜索能力较强的人工鱼群算法对模型求解,搜索出部署模型的全局最佳值,从而降低了云资源部署时的容器负载。

(4) 容器云资源平均能量损耗测试。

采用方法 1、方法 2、方法 3 以及方法 4,对 24 小时内的容器云资源平均能量损耗进行测试,测试结果如表 3 所示。

根据表 3 中的数据可知,经过多次实验,方法 1 的容器云资源平均能量损耗在 25 672 kJ 以内;方法 2、3、4 的容器云资源平均能量损耗均高于 30 000 kJ,高于方法 1,方法 1 的容器云资源平均能量损耗最低,说明利用该文提出的基于人工鱼群算法的容器云资源低能耗部署方法进行容器云资源低能耗部署的效果最好。

表 3 不同部署方法的容器云资源平均能量损耗测试结果

实验次数/次	平均能量损耗/kJ			
	方法 1	方法 2	方法 3	方法 4
2	25 672	30 512	35 205	36 052
4	24 662	31 568	35 689	36 925
6	23 156	31 682	36 054	37 020
8	24 952	32 895	36 854	38 205
10	24 856	33 965	37 825	39 054

综上所述,方法 1 在开展容器资源部署方法测试过程中,测试出的部署效果高于其他方法,证明方法 1 在资源部署时,具备有效性。

4 结束语

随着国家节能减排的要求增多,使容器云资源低能耗部署方法变得尤为重要。针对传统资源部署方法中存在的问题,提出基于人工鱼群算法的容器云资源低能耗部署方法。该方法通过容器云资源配置分析结果,建立云资源的低能耗部署模型;然后使用人工鱼群算法获取模型最佳值,建立最佳部署策略,完成容器云资源的低能耗部署。由于该方法在选取容器运行参数时存在误差,今后会针对该项缺陷,继续对该部署方法实施优化处理。

参考文献:

[1] LI J, QIAO Z, ZHANG K, et al. A lattice-based homomorphic proxy re-encryption scheme with strong anti-collusion for cloud computing[J]. Sensors, 2021, 21(1): 288.
 [2] WANG J, XING C. Factors affecting the evolution of ad-

- vanced manufacturing innovation networks based on cloud computing and multiagent simulation[J]. *Mathematical Problems in Engineering*, 2021, 2021(1):1-12.
- [3] ZHENG J, ZHANG Z, MA Q, et al. Multi-resource VNF deployment in a heterogeneous cloud[J]. *IEEE Transactions on Computers*, 2022, 71(1):81-91.
- [4] HAN X, MENG X, YU Z, et al. A service function chain deployment method based on network flow theory for load balance in operator networks[J]. *IEEE Access*, 2020, 8:93187-93199.
- [5] 龚坤, 武永卫, 陈康. 容器云多维资源利用率均衡调度研究[J]. *计算机应用研究*, 2020, 37(4):1102-1106.
- [6] 徐胜超, 熊茂华. 基于遗传算法的容器云资源配置优化[J]. *计算机与现代化*, 2022(1):108-112.
- [7] 李启锐, 彭志平, 崔俊龙, 等. 容器云环境虚拟资源配置策略的优化[J]. *计算机应用*, 2019(3):784-789.
- [8] WANG X. Recognition and positioning of container lock holes for intelligent handling terminal based on convolutional neural network[J]. *Traitement du Signal*, 2021, 38(2):467-472.
- [9] CUONG T N, KIM H S, XU X, et al. Container throughput analysis and seaport operations management using nonlinear control synthesis [J]. *Applied Mathematical Modelling*, 2021, 100:320-341.
- [10] POTDAR A M, NARAYAN D G, KENGOND S, et al. Performance evaluation of docker container and virtual machine [J]. *Procedia Computer Science*, 2020, 171:1419-1428.
- [11] VALDEZ M G, GUERVÓS J J M. A container-based cloud-native architecture for the reproducible execution of multi-population optimization algorithms [J]. *Future Generation Computer Systems*, 2021, 116(1):234-252.
- [12] TAN B, MA H, MEI Y, et al. A cooperative coevolution genetic programming hyper-heuristic approach for on-line resource allocation in container-based clouds[J]. *IEEE Transactions on Cloud Computing*, 2022, 10(3):1500-1514.
- [13] BRAIKI K, YOUSSEF H. Fuzzy-logic-based multi-objective best-fit-decreasing virtual machine reallocation [J]. *Journal of Supercomputing*, 2020, 76(1):427-454.
- [14] GHASEMI A, HAGHIGHAT A T. A multi-objective load balancing algorithm for virtual machine placement in cloud data centers based on machine learning [J]. *Computing*, 2020, 102(9):2049-2072.
- [15] KARTHIKEYAN L, VIJAYAKUMARAN C, CHITRA S, et al. SALDEFT: self-adaptive learning differential evolution based optimal physical machine selection for fault tolerance problem in cloud [J]. *Wireless Personal Communications*, 2021, 118(2):1453-1480.
- [16] ZHANG Q, TANG C, BAI T, et al. A two-layer optimal scheduling framework for energy savings in a data center for cyber-physical-social systems [J]. *Journal of Systems Architecture*, 2021, 116(2):102050.
- [17] ZHANG F, ZHANG J, NI H, et al. Optimization of Al-Si10MgMn alloy heat treatment process based on orthogonal test and grey relational analysis [J]. *Crystals*, 2021, 11(4):385.
- [18] ZHANG L, FU M, LI H, et al. Improved artificial bee colony algorithm based on damping motion and artificial fish swarm algorithm [J]. *Journal of Physics: Conference Series*, 2021, 1903(1):012038.
- [19] ZHANG L, FU M, FEI T, et al. The artificial fish swarm algorithm optimized by RNA computing [J]. *Automatic Control and Computer Sciences*, 2021, 55(4):346-357.
- [20] 何力, 吕红芳, 李俊甫, 等. 基于模拟退火算法改进的人工鱼群算法(SA_AFSA)的微电网能量优化调度研究[J]. *太阳能学报*, 2020, 41(9):36-43.
- [21] 王丽, 王威. 基于鱼群算法的高光谱图像稀疏分解研究[J]. *计算机仿真*, 2020, 37(1):226-233.
- [22] 张朝炜, 柳云祥, 朱永利. 基于改进人工鱼群算法的大规模多目标机组组合优化[J]. *电力系统保护与控制*, 2021, 49(8):100-108.
- [23] 郭伟, 秦国选, 王磊, 等. 基于改进人工鱼群算法和MAKLINK图的机器人路径规划[J]. *控制与决策*, 2020, 35(9):2145-2152.