

# 容器云平台中基于 Actor-Critic 的深度强化学习任务调度方法

韩卓

江南机电设计研究所 贵州 贵阳 550009

**【摘要】**：为了改善容器云平台传统任务调度算法所存在的资源分配不均、动态适应性差、多目标优化冲突等状况，提出了一种用 Actor-Critic 深度强化学习来实现的任务调度方法。本方法依靠深度强化学习的自主决策和动态优化能力，创建容器云集群状态感知模型，把任务调度变成序列决策问题，用 Actor 网络输出调度决策，用 Critic 网络实时评价决策价值，从而达成调度策略的端到端自主优化。从实验结果可以看出，该方法相比传统的调度算法，在资源利用率、任务完成时延、集群负载均衡度等主要指标上都有明显提高，可以很好地适应容器云平台动态多变的业务环境，为容器云高效任务调度提供了一种可行的技术方案。

**【关键词】**：容器云平台；任务调度；Actor-Critic；深度强化学习；资源优化

DOI:10.12417/3041-0630.26.05.043

## 1 引言

伴随着云计算和微服务架构的迅速发展，容器云平台由于轻量级、弹性伸缩、部署高效等特点，成为了云计算基础设施的主要载体。Actor-Critic 算法是深度强化学习的一种经典结构，它有策略梯度和值函数优化的优点，克服了传统强化学习样本效率低、收敛慢的问题，可以对连续的状态和动作空间进行高效的决策。本文以 Actor-Critic 架构为基础，设计出适合于容器云平台的任务调度模型，建立多目标优化奖励函数，使资源利用率、任务时延、负载均衡三者达到协同优化的目的，从而提高容器云平台整体运行效率和服务稳定性。

## 2 相关理论基础

### 2.1 容器云任务调度核心需求

容器云任务调度的主要目的就是把 arriving 的任务合理地分发到集群中的各个计算节点上，保证集群资源的充分利用以及业务服务质量。调度过程中要同时考虑三个主要的指标，即资源利用率，最大限度地提高 CPU、内存、带宽等资源的利用率，降低由于资源浪费造成的损失；任务执行效率，加快任务排队和完成的时间，保证业务的实时性；负载均衡度，防止单个节点的资源过度使用导致服务故障，使集群整体稳定。传统的调度方法不能同时满足多目标协同优化的要求，容易造成某个指标最优但是整体性能较差的情况。

### 2.2 Actor-Critic 深度强化学习原理

Actor-Critic 算法由 Actor 网络和 Critic 网络两部分组成，两者一起完成决策和评价的过程。Actor 网络属于策略网络，感知环境状态并输出相应的调度动作，利用策略梯度更新网络参数，逐步改善决策策略；Critic 网络属于值网络，接收环境状态和 Actor 输出的动作，计算状态动作价值函数，实时评价当前决策的好坏，给 Actor 网络参数更新提供梯度指导。

该架构不同于传统的值函数强化学习和策略梯度强化学习，不需要等整个任务序列完成之后再单步更新，收敛速度更快，适合容器云实时调度场景。

## 3 基于 Actor-Critic 的容器云任务调度模型设计

### 3.1 调度问题建模

把容器云集群当作强化学习交互环境，调度代理对应 Actor-Critic 智能体，任务调度过程变成智能体和集群环境的连续交互过程。定义集群状态空间、动作空间、奖励函数这三个主要要素，完成调度问题的数学建模。

状态空间是由集群全局状态和节点局部状态组成，包含各个节点的 CPU 利用率、内存利用率、带宽占用率、待执行任务队列长度、任务资源需求等各个方面，对容器云集群实时运行状态进行全方面的描述。动作空间是指任务和计算节点之间的匹配关系，即智能体对当前需要调度的任务进行选择，然后选取最合适的执行节点，最后给出一个离散的调度决策。

奖励函数是调度策略优化的主要方向，用多目标加权融合的方式设计，既考虑资源利用率又考虑任务时延，也考虑负载均衡度。奖励值与资源利用率、负载均衡度成正比，任务完成时延与奖励值成反比，智能体为了得到最优调度方案，会综合考虑各个指标的影响。

### 3.2 模型整体架构

本文所设计的调度模型分为状态感知层、智能决策层和执行反馈层三个层次。状态感知层对容器云集群节点和任务数据进行实时采集，对数据进行预处理和特征提取，得到标准化的状态向量输入到智能决策层。智能决策层为中心层，主要包含 Actor 网络和 Critic 网络，Actor 网络用全连接神经网络提取状态特征并输出任务调度节点决策，Critic 网络同时接收状态向量和 Actor 决策结果，计算状态价值函数来评价决策合理性，

并生成误差信号，反向传播更新两个网络的参数。

### 3.3 模型训练与优化流程

模型训练分为离线预训练和在线微调两个阶段。离线预训练阶段利用容器云仿真集群环境生成大量的模拟任务和节点数据，使智能体进行初始策略的学习，快速收敛到基础的最优策略，避免在线训练对真实集群运行造成影响。在线微调阶段将预训练好的模型部署到真实的容器云平台中，根据实时的业务流量、集群状态等信息对网络参数进行小范围调整，以适应真实的环境变化，保证调度策略的实时性、适应性。

## 4 实验与结果分析

### 4.1 实验环境设置

搭建容器云仿真测试平台，集群有8个异构计算节点，节点CPU核心数、内存容量不同来模拟真实的生产环境异构特性。设置5个不同的负载强度的测试场景，任务类型有计算密集型、内存密集型和混合型，包含日常负载、峰值负载等各种业务场景。对比算法选择传统的轮询算法、最小负载优先算法、遗传算法，用资源利用率、平均任务完成时延、集群负载均衡度作为评价指标来检验本文的方法是否有效。

### 4.2 实验结果与对比

实验对各个算法在不同的负载情况下的主要指标进行了统计，结果见表1。数据是5组场景的平均值。

调度算法	平均CPU利用率(%)	平均内存利用率(%)	平均任务完成时延(ms)	负载均衡偏差值
轮询算法	62.3	58.7	186.4	0.28
最小负载优先算法	71.5	67.2	153.2	0.19
遗传算法	75.8	72.4	131.7	0.15
本文 Actor-	82.6	79.3	98.5	0.08

### 参考文献:

- [1] 张玉樟,田乐,魏华利,等.基于策略熵监督的在线强化学习任务调度算法[J/OL].计算机工程,1-13[2026-03-23].<https://doi.org/10.19678/j.issn.1000-3428.0253414>.
- [2] 周雅,王乾,方如举.配电网中基于混合DRL的任务卸载与多资源协同调度优化方法[J].电力系统保护与控制,2026,54(04):165-174.
- [3] 陈永,刘骅驹,张冰旺.基于深度强化学习的高速铁路监控视频MEC智能卸载方法[J].铁道学报,2026,48(02):96-104.

Critic 方法				
-----------	--	--	--	--

由实验数据可知，本文所提出的一种基于 Actor-Critic 的深度强化学习调度方法，在各项指标上都比传统的调度算法好。平均 CPU 利用率、内存利用率分别为 82.6%、79.3%，比轮询算法提高 20% 以上，比遗传算法提高近 7%，大大减少了集群资源的闲置率。平均任务完成时延为 98.5ms，比传统的算法缩短了 30% 以上，业务执行效率得到提高。负载均衡偏差值越小，说明集群中各个节点的负载分布越均匀，可以有效地防止节点过载的问题，提高集群的运行稳定性。本文的方法在高负载峰值的时候更具有优势，可以快速地对任务流量的变化作出反应，并且可以动态地调节调度策略，保证集群的稳定运行。

### 4.3 算法性能分析

本文的方法以 Actor-Critic 架构为基础，双网协同来实现集群状态与调度决策之间映射的学习，不需要人为调节参数，适合于容器云动态变化的业务场景。相比传统的启发式算法，该方法具有更强的自适应能力和全局优化能力，不容易陷入局部最优的陷阱，相比于单一的强化学习算法，收敛速度快，调度决策的实时性好，可以满足容器云在线调度的要求。多目标奖励函数的设计可以达到资源效率、业务时延、集群稳定性三者之间的平衡，更加符合生产环境实际调度的要求。

## 5 结论

本文针对容器云平台传统的任务调度算法存在的问题，提出了一个基于 Actor-Critic 的深度强化学习的任务调度方法，用强化学习来对任务调度进行自主决策以及动态优化。实验结果表明，该方法可以有效地提高容器云资源利用率、降低任务执行时延、改善集群负载均衡、整体性能明显好于传统的调度算法。