

doi: 10.7690/bgzdh.2025.03.007

边缘计算网络中多核任务卸载调度和资源适配研究

李 金¹, 樊腾飞¹, 高红亮¹, 刘科孟¹, 谢 虎²

(1. 中国南方电网电力调度控制中心自动化处, 广州 510530;

2. 南方电网数字电网研究院有限公司数字电网分公司, 广州 510560)

摘要: 为解决边缘计算网络任务卸载中的问题, 对移动边缘关键技术进行研究。设计边缘节点计算分布式架构, 参考量子粒子群算法和容器技术, 形成基于边缘网关架构的任务卸载优化策略; 对优化策略进行仿真实验, 通过改变计算任务规模以及计算任务大小, 分析任务卸载时延和耗能。结果表明: 该策略能够有效降低任务卸载时延和耗能, 实现边缘节点资源的充分利用, 达到资源的良好适配效果。

关键词: 边缘节点; 边缘计算集群; 分布式架构; 任务卸载; 资源适配

中图分类号: TP393 **文献标志码:** A

Research on Multi-core Task Offload Scheduling and Resource Adaptation in Edge Computing Networks

Li Jin¹, Fan Tengfei¹, Gao Hongliang¹, Liu Kemeng¹, Xie Hu²

(1. Automation Department, China Southern Power Grid Power Dispatching Control Center,

Guangzhou 510530, China; 2. Digital Power Grid Branch, Digital Grid Research Institute,

China Southern Power Grid, Guangzhou 510560, China)

Abstract: In order to solve the problem of network task unloading in edge computing, the key technologies of mobile edge are studied. Design a distributed architecture for edge node computing, referencing quantum particle swarm optimization algorithm and container technology, to form a task offloading optimization strategy based on edge gateway architecture; Conduct simulation experiments on the optimization strategy, analyze task offloading latency and energy consumption by changing the scale and size of computing tasks. The results show that this strategy can effectively reduce task offloading latency and energy consumption, achieve full utilization of edge node resources, and achieve good resource adaptation effect.

Keywords: edge node; edge computing cluster; distributed architecture; task offloading; resource adaptation

0 引言

工业物联网近年来取得了长足发展。生产生活的方方面面运用的智能终端设备越来越多, 呈现爆发式增长, 给通信网络以及云计算带来了巨大的挑战。基于云计算集中式的计算模式已经无法满足物联网数据计算低时延、可靠性高的需求。把全部或者部分需要密集计算的工作, 卸载到网络边缘计算节点上的相关边缘计算服务也就逐步出现, 成为了当下研究的热点^[1]。就边缘计算而言, 一般都采取诸多边缘节点对计算任务进行协同处理, 其核心是合理调度、高效分配边缘节点的相关计算资源。多数终端的计算和存贮资源都非常有限, 移动边缘计算网络能够在更接近终端设备的场所部署存贮、控制、联网、计算等功能, 距离用户端比较近^[2]。将边缘计算服务器、边缘计算网关、边缘计算节点等

边缘设备引入进来, 边缘节点不同, 它们的功能也各不相同^[3]。

Yu^[4]提出一种基于区块链和边缘计算的分布式可信认证系统, 目的在于提高物联网终端节点的认证效率。毛水强等^[5]采用一种基于深度强化学习的深度确定性策略梯度算法, 将移动边缘技术任务处理问题转化为马尔科夫决策过程来降低任务卸载时延。Pedro 等^[6]在边缘计算环境下, 针对用户间跨域认证与密钥协商问题, 利用区块链技术以及口令的简便性, 加入生物特征唯一性, 设计跨域协议, 实现了用户在边缘认证服务器和区块链处的双重认证。王伏亮等^[7]运用边缘计算和机器学习技术, 采用拓扑排序、决策树等方式寻求任务相关性, 提出了一种工业智能边缘计算中基于数据流相关性的计算卸载方法, 以解决存储器容量不足等问题。移动边缘计算网络系统中网络的时变性和任务生成的动

收稿日期: 2024-07-09; 修回日期: 2024-08-12

基金项目: 南方电网公司重点科技项目(0000002021030101XT00045)

第一作者: 李 金(1979—), 男, 甘肃人, 硕士。

态性, 为求解任务卸载问题带来了挑战。边缘卸载任务存在着诸多问题, 计算任务的动态性与超密集网络下大量节点之间相互影响。已有的计算卸载研究大多局限于普通的静态场景中, 既不能满足实现动态的适用多场景的计算卸载, 又不能满足边缘网关接入云大脑的可靠性和安全性的需求。

因为应用多样性, 不能均匀地分布设备等, 造成了任务卸载到相关边缘服务器之间的任务量存在着较大差异, 边缘计算发生负载不均衡的情况很可能出现^[8]。如果计算模式采取动态分布式, 既可以使计算任务到计算资源的距离更近, 又能够在一定程度上控制数据传到云平台的传输时延效果, 也可以在传输过程中降低可能发生的安全问题, 具有更多的存储功能, 可靠性能更高^[9]。

为解决边缘计算网络任务卸载中存在的问题, 在移动边缘技术模型上进行改进, 针对关键技术进行研究, 设计边缘节点计算分布式架构, 参考量子粒子群算法, 引入容器技术的任务卸载优化策略, 就优化策略开展仿真实验, 卸载调度边缘计算网络多核任务以及资源进行适配等。使任务下载技术更加符合物联网的需求, 能够给边缘计算网络边缘节点技术提供参考。

1 材料与方法

首先, 对相关理论核心知识进行说明, 包括移动边缘计算的定义和架构、云边协同计算分布式架构、量子粒子群算法及多容器技术; 其次, 进行了边缘网关架构的优化; 最后, 设计了仿真实验。

1.1 理论基础

1.1.1 移动边缘计算定义和架构

移动边缘计算(mobile edge computing, MEC)是从核心网络将远端的云计算能力拓展到相应的网络边缘, 增强在网络边缘部位的联合存储、计算以及通信等核心能力, 可以缓解中心云负载压力, 与用户的距离更近。MEC最大的特点是将远端的计算和存储中心下沉到靠近用户的地方, 这样可以近距离地响应用户的请求, 增强移动终端的处理能力。边缘计算大致涵盖了终端、边缘和云 3 层基础架构^[10]。云层也称之为中心云, 边缘层可以由多个 MEC 服务器构成, 终端层由各种终端设备组成。不同层级间能够彼此通信, 每层所处的位置, 决定了该层的存储以及计算能力, 对各层级的功能具有决定性作用。图 1 是详细的边缘计算架构。

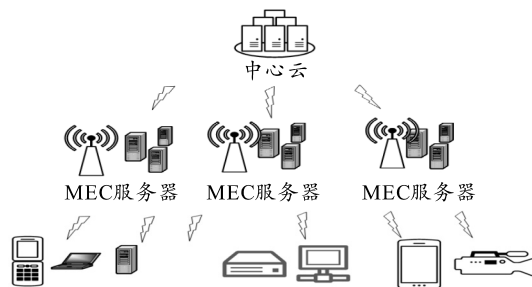


图 1 边缘计算架构

图 1 中这种自上而下的模式, 由中心云进行任务分割, 并将任务分发到移动边缘节点 MEC 服务器上, 通过 MEC 服务器将任务卸载到边缘设备中。移动边缘节点服务器负责边缘节点以及终端设备, 如任务卸载、资源管理、任务调度、网关管理、应用管理等统一管理。

1.1.2 云边协同计算分布式架构

在分布式节点系统中, 边缘云负责与边缘设备进行数据传输、节点计算、任务执行等功能的通信。移动设备可以直接与附近的边缘云节点交互, 不需要与中心云交互, 这可以提高边缘设备的时效性, 节省云的流量带宽。边缘节点分布式架构是由多个计算机服务器组成的系统。计算机服务器称为节点, 节点之间相互连通, 节点与上层和下层之间也相互连通。这些连通的节点相互之间的操作会有一定协同作用。中心云数据中心与边缘节点的调度为自上而下的模式。云端将需要依赖大量资源进行计算的任务进行分割, 将分割后的任务分发到边缘服务器上, 最终卸载到终端设备, 充分利用终端设备的计算能力, 从而降低中心云的计算压力, 降低整个系统的耗能和传输延时。分布式服务最大的优点是拓展性和稳定性, 边缘节点模型能够很好管理所有节点并根据相应参数将任务分配到更合适的节点上。其主要部分为边缘计算节点和终端设备。在分布式节点系统中, 云端负责和边缘设备进行数据传输、节点计算、任务执行^[11]。具体计算架构如图 2 所示。

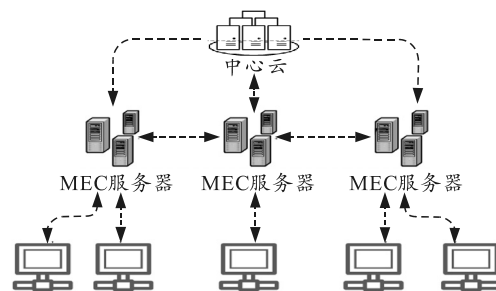


图 2 分布式节点组织架构

图2中,中心云可以向各个边缘节点发送数据,每个边缘节点集群也可以向一层的云数据中心传输数据。而且每个边缘节点集群可以相互传输数据。系统中包含的边缘节点服务器,对外展示为单个的边缘节点。移动终端收集、处理完数据后,可以向边缘节点发送请求,边缘节点也根据任务调度策略,将任务发送到更合适的边缘节点。

1.1.3 量子粒子群算法

QPSO算法则是在PSO算法的基础上,通过量子力学的方式进行改进的算法^[12]。该算法能够适当降低搜索范围,不会将搜索范围扩散到无穷大。只是在整个空间中搜索满足聚集性质的粒子,可以通过量子粒子群算法来解决边缘节点的资源调度问题^[13]。可以获得局部最优的解决方案。调度中心负责收集目标节点的负载状况,用 φ_k 表示^[14]。卸载节点的计算资源 f_{ik} 计算方法如式(1)所示:

$$f_{ik} = \frac{a_{ik} \sum_{ij} s_i}{\varphi_k + a_{ik} \sum_{ij} s_i} f_{\max} \quad (1)$$

假设粒子群为 n ,则粒子对应的位置向量为:

$$\lambda_n = \{\lambda_{n1}, \lambda_{n2}, \dots, \lambda_{nk}\} \quad (2)$$

经过 m 次迭代后,第 n 个粒子的最佳位置为 $p_n(m)$,当前粒子的最佳位置为 $G(M)$,在进行下一次迭代时,最佳位置 P 的表达式如式(3)所示:

$$P = \varphi p_n(m) + (1 - \varphi)G(m) \quad (3)$$

1.1.4 多容器技术

容器是信息化应用程序,基于Go语言设计而成,它对应用程序组件的生命周期进行管理,使操作环境以及应用程序密切融合,可以部署在任何地方且有效运行。容器技术日益成熟,在物联网边缘云得到了广泛应用。Docker容器是一款容器引擎,与传统虚拟技术相比,具有很多优势^[15],Docker容器可以在秒级乃至毫秒级即可实现操作命令,不会占用较高的资源比例^[16]。一台服务器上通常能够部署千百计的容器,因此形成的虚拟化集群非常庞大。可将其分为客户端、服务器、仓库3个模块。在容器云中,任务调度算法的实际性能,会对容器集群的整体性能产生直接影响,客户端请求计算任务卸载的情况下,边缘计算平台服务器能够按照客户终端的相关请求规模动态,开展容器分配工作^[17]。容器云平台可以给客户提供充足的计算资源,在满足客户计算需求的情况下,可以实现低时

延任务下载。

1.2 边缘网关架构设置

基于1.1小节的内容,进行边缘网关架构的计算。在分布式边缘计算技术架构的基础上,参考量子粒子群算法,引入容器技术的任务卸载优化策略,尽最大限度用好边缘节点的相关计算资源,以合理地调度任务卸载,该边缘计算可以应用于更多的物联网场景。边缘网关功能更加齐全、边缘节点安全防护性能更高^[18]。详细的组织架构如图3所示。

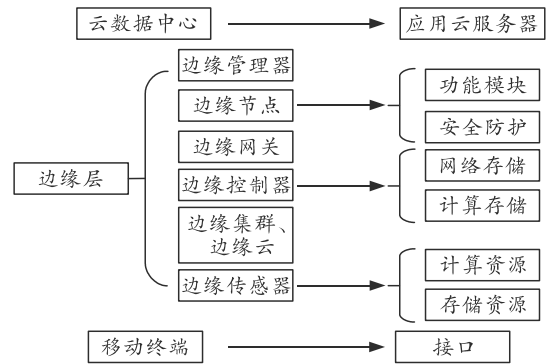


图3 边缘计算设计架构

该架构从上到下分为云数据中心、边缘和移动终端3层。在云数据中心与移动终端中间的边缘层,在架构整体中居于核心地位,它涵盖了2部分,分别是边缘管理器和边缘节点。作为硬件实体,边缘节点在边缘计算居于核心地位。为适用不同场景下的需求,边缘节点涵盖了多种边缘,例如边缘网关、边缘传感器、边缘集群、边缘控制器、边缘云等,存在着相当多的功能模块,例如满足设备接入与数据采集服务、安全管理服务、资源管理、应用管理。能够实现边缘节点的安全防护,如边缘基础设施安全、边缘网络安全、边缘数据安全、边缘应用安全、边缘安全生命周期管理以及边云协同安全。

假设 w 为执行任务,则调度到异地的任务可以表示为:

$$w_i^j = F \times w_i^o \quad (4)$$

式中: F 为调度节点任务的执行速度与本地服务器的执行速度的正比值; i 为节点序号; j 为数据序号; O 为输出。

在过去的研究中,只考虑了节点簇的简单划分。然而,在云边缘协作系统中,任务不仅将在边缘云中心执行,而且卸载到其他边缘节点。如果边缘服务已完全加载,它将被调度到其他边缘服务器。因此,需要增加节点之间的通信和传输成本。其次,节点之间的传输带宽不完全相同,即服务节点之间

的通信成本与调度中心的通信成本不相同。实际上,不同的云和边缘节点网络是不同的,因此需要不同的计算。本地服务和远程服务器之间的节点任务的传输时间可以表示:

$$T_{\text{offer}}^i = \frac{A}{B} \quad (5)$$

式中: A 为调度该任务所需的数据量; B 为传输带宽; T 为节点间数据传输所用时间。

定时任务调度总传输时间代价如式(6)所示:

$$w_{\text{offer}}^i = \frac{\text{in}_{ij}}{B} + \frac{\text{out}_{ij}}{B} \quad (6)$$

任务的总时间代价由任务调度代价、执行时间代价、通信传输时间代价共同构成,如式(7)所示:

$$T_{\text{total}} = T_{\text{local}} + T_{\text{transfer}} + T_{\text{offer}} \quad (7)$$

式中: T_{transfer} 为向边缘设备传输数据需要时间; T_{total} 为本地计算时间; T_{offer} 为边缘节点计算任务需要时间。

顶点集 V 中的一个节点 i 的所有卸载任务通信时延为:

$$\sum_{mn} \frac{x_{mn}(a_{ik})a_{ik} \sum_{lj} s_{ij}}{r_{mn}} \quad (8)$$

式中: $a_{ik} \sum_{lj} s_{ij}$ 为节点 v_i 任务卸载到节点 v_k 上的数据量; a_{ik} 为边缘节点 v_i 卸载到边缘节点 v_k 上的任务占总任务比例; r_{mn} 为节点间链路传输速率,用于区分 a_{ik} 是否通过这条链路传输,一般取值为 1 或 0。

假设节点 v_k 分配给节点 v_i 的可用算力为 f_{ik} , 则 $\sum_{k=1}^i f_{ik} \leq f_k^{\max}$ 。处理时延由计算时延和传输时延组成,则总时延为:

$$\sum_{mn} \frac{x_{mn}(a_{ik})a_{ik} \sum_{lj} s_{ij}}{r_{mn}} + \frac{a_{ik} \sum_{lj} s_{ij}}{f_{ik}} \quad (9)$$

式中 $k=i$ 时, 时延最小。

如果边缘云中心仅选择最近的节点卸载,则相邻节点也可能处于满载状态,从而导致任务的二次拥塞。执行时间类似于或甚至比本地执行更差。此外,多个节点还可能同时将任务卸载到一个节点,导致卸载的节点过载,这将导致多个任务的二次拥塞。基于此类问题,云边协同系统的过载节点需要卸载的任务选择如式(10):

$$\min_{a,x,f} \max_k \left\{ \sum_{mn} \frac{x_{mn}(a_{ik})a_{ik} \sum_{lj} s_{ij}}{r_{mn}} + \frac{a_{ik} \sum_{lj} s_{ij}}{f_{ik}} \right\} \quad (10)$$

卸载条件如式(11):

$$\sum_{mn} \frac{x_{mn}(a_{ik})a_{ik} \sum_{lj} s_{ij}}{r_{mn}} + \frac{a_{ik} \sum_{lj} s_{ij}}{f_{ik}} \leq \frac{\sum_{lj} s_{ij} c_i}{f^{\max}} \quad (11)$$

结合 1.1 节中的量子群算法内容,进行边缘节点资源调度问题的解决。综上所述,笔者构建算法的总体步骤可以总结为 3 方面: 1) 获取可执行节点、测算网络状态、代价计算。即首先进行节点的划分,得到节点的可使用算力和权值。2) 对节点的网络状态进行测算,清楚异常节点,对节点权值变化进行计算。3) 对节点通信时间代价进行计算,并通过总分数对比得到最优节点。

1.3 仿真实验

多核任务卸载调度优化策略通过引入分布式技术和容器技术,对边缘集群和边缘网关进行弹性接入,充分利用边缘节点的计算资源,实现计算能力的快速扩展和资源利用率的提升。

针对以多容器技术为基础的边缘计算任务,设计了 3 组卸载调度策略的实验。

- 1) 调整计算任务规模,在卸载调度策略不一样的情况下,测试相关计算任务卸载影响时延的情况。
- 2) 调整计算任务规模,在卸载调度策略不同的情况下,测试计算任务卸载影响能耗的情况。
- 3) 调整计算任务的大小,在卸载调度策略不同的情况下,测试计算任务卸载影响时延的情况。

搭建了基于 Docker 容器的边缘云环境,以边缘节点为基础,依次部署内存密集型、网络密集型、计算密集型的容器。对任务下载云端平台进行设置,模拟多个终端进行计算任务卸载。测试不同卸载调度策略下的任务下载时延和耗能分析。

2 结果与分析

2.1 计算任务规模对计算任务卸载的时延的影响

如图 4 所示,当计算任务规模发生变化,卸载调度策略不同,影响计算任务卸载时延的情况也不同。

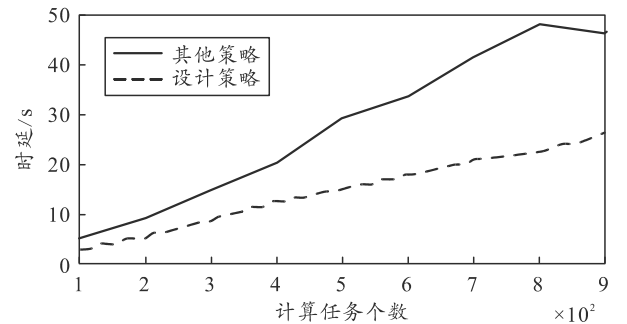


图 4 不同策略下不同计算任务规模的时延

从实验结果中能够看出：与其他策略相比，笔者的设计策略在执行计算任务卸载的过程中，时延降低幅度很大，特别是计算任务数达到 900 个的情况下，其他策略以及设计策略时延分别是 47 s 以及 26 s，比其他策略的时延下降了 46.4%；因此，与其他策略相比，这种设计策略在数量规模不同的情况下，其计算任务卸载能够在较大程度上降低时延。

2.2 计算任务规模对计算任务卸载的耗能影响

改变计算任务规模，不同卸载调度策略下进行的计算任务卸载对耗能的影响如图 5 所示。

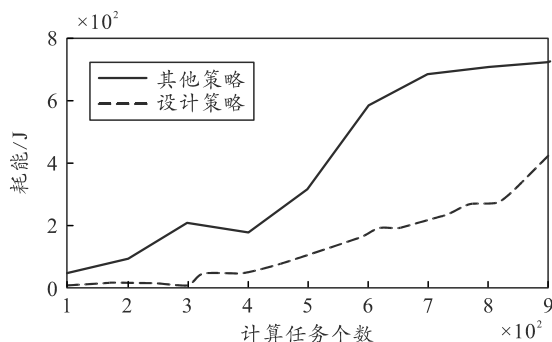


图5 不同策略下不同计算任务规模的能耗

就实验结果而言，能够注意到与其他策略相比，所论述的设计策略可以显著地削减计算任务卸载的相关能耗，随着任务数量增多，双方的差距也会越来越大，在计算任务数达到 900 个的情况下，其他策略耗能约为 795 J，设计策略耗能为 400 J，提出的设计策略相比于其他策略能耗降低了 49.7%。不同数量规模的计算任务卸载的能耗有大幅度地降低。

2.3 计算任务大小对计算任务卸载的时延影响

改变计算任务大小，不同卸载调度策略下进行的计算任务卸载对时延的影响如图 6 所示。

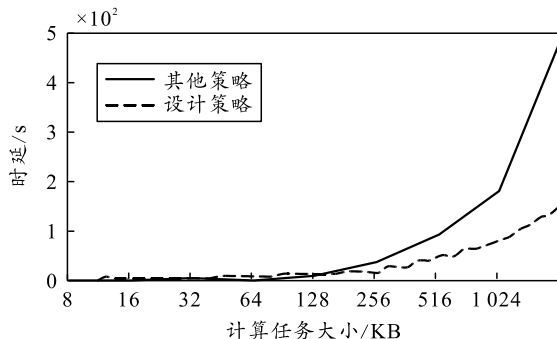


图6 不同策略下不同计算任务大小的时延

从实验结果中可以发现：提出的设计策略随着任务大小增加任务卸载延时逐渐降低，当计算任务达到 1 024 KB 时候，2 种策略之间的延时开始出现

明显的差别，计算任务为 2 048 KB 时，达到了最大差距，与其他策略相比，设计策略的时延减少了 69.5%。与其他策略相比，所提设计策略能够结合计算任务卸载的情况，其时延降低幅度很大。

3 结论

笔者在移动边缘计算 MEC 的基础上，对其关键技术进行研究和改进，设计边缘节点计算分布式架构，参考量子粒子群算法，引入容器技术的任务卸载优化策略，优化边缘网关架构的相关设置。针对优化策略实施仿真实验，在容器部署环境下，通过改变计算任务规模以及计算任务大小，探究边缘计算任务卸载优化调度策略下进行的任务卸载对时延以及耗能的影响。结果表明：提出的边缘计算任务卸载优化调度策略相比其他策略，能有效降低任务下载时延和耗能，实现边缘节点资源的充分利用，达到资源的良好适配效果，获得优越的性能提升。

参考文献：

- [1] 刘婷, 罗喜良. 移动边缘计算中的在线任务卸载方法[J]. 中国科学院大学学报, 2022, 39(2): 267-274.
- [2] ALEKSANDROVA E B, OBLOGINA A Y, SHKORKINA E N. Authentication of Control Devices in the Internet of Things with the Architecture of Edge Computing[J]. Automatic Control and Computer Sciences, 2022, 55(8): 1087-1091.
- [3] 郑守建, 彭晓晖, 王一帆, 等. 一种基于综合匹配度的边缘计算系统任务调度方法[J]. 计算机学报, 2022, 45(3): 485-499.
- [4] YU J. Special issue on Intelligent software services for IoT and edge computing[J]. Journal of Reliable Intelligent Environments, 2022, 8(1): 1.
- [5] 毛水强, 洪健, 任华, 等. 面向电力物联网的 5G 移动边缘计算任务卸载方法[J]. 电测与仪表, 2022, 59(2): 105-111.
- [6] PEDRO R J, SALVADOR A, KATJA G, et al. Arithmetic Framework to Optimize Packet Forwarding among End Devices in Generic Edge Computing Environments[J]. Sensors, 2022, 22(2): 421.
- [7] 王伏亮, 葛永高, 李澄, 等. 基于边缘计算技术的融合终端 APP 系统设计[J]. 电子设计工程, 2022, 30(2): 165-169.
- [8] JÁNOS H, MARKOSZ M. Architecture Integration of 5G Networks and Time-Sensitive Networking with Edge Computing for Smart Manufacturing[J]. Electronics, 2021, 10(24): 3085-3085.
- [9] 张利, 刘洛川, 李光. 一种基于边缘计算的分布式高精度定位方法[J]. 长江信息通信, 2022, 35(2): 34-42.
- [10] SONG S D. Delay-sensitive tasks offloading in

- multi-access edge computing[J]. Expert Systems With Applications, 2022, 198: 116730.
- [11] 胡忠波, 贾芝婷, 王素贞. 边缘计算关键技术分析及应用平台比较研究[J]. 河北省科学院学报, 2021, 38(5): 41-49.
- [12] XING X. An Improved Quantum Particle Swarm Algorithm for Routing Optimization of Wireless Sensor Networks[J]. International Journal of Circuits, Systems and Signal Processing, 2021, 15: 33-39.
- [13] 张艺瀛, 金志刚. 一种高维多模态优化的量子粒子群优化算法[J]. 哈尔滨工业大学学报, 2018, 50(11): 50-58.
- [14] GROTTI E. A novel multi-objective quantum particle swarm algorithm for suspension optimization[J]. Computational and Applied Mathematics, 2020, 39(4): 523-532.
- [15] 陈文, 胡春潮, 尤毅, 等. 基于容器技术的营配一体化智能终端软件架构与实现机制[J]. 电工技术, 2022(3): 149-152.
- [16] VERMA R K, GUPTA R. Efficient Docker Container Scheduling using ABC Optimization Technique[J]. International Journal of Innovative Technology and Exploring Engineering, 2019, 8(9s4): 26-32.
- [17] 杨漾, 敖知琪, 刘佳, 等. 面向数字电网的基于容器技术的边缘计算数据处理机制[J]. 南方电网技术, 2021, 15(5): 98-103.
- [18] XU R X. Distributed Secure Edge Computing Architecture Based on Blockchain for Real-Time Data Integrity in IoT Environments[J]. Actuators, 2021, 10(8): 197-197.

(上接第 12 页)

参考文献:

- [1] 肖俊. ECDIS 软件人机界面可用性评价[J]. 科研管理, 2016, 37(S1): 78-84.
- [2] 叶坤武, 包涵, 魏思东. 基于视觉注意力分配的飞机驾驶舱人机界面布局优化[J]. 南京航空航天大学学报, 2018, 50(3): 416-421.
- [3] 谢伟, 辛向阳. 基于无意识认知的交互设计研究[J]. 现代电子技术, 2016, 39(12): 22-25.
- [4] 谢伟, 辛向阳, 丁静雯. 基于眼动测试的产品人机界面交互设计研究[J]. 机械设计, 2015, 32(12): 110-115.
- [5] 张宇红, 朱昊. 基于眼动仪的智能冰箱人机界面评估方法研究[J]. 包装工程, 2019, 40(24): 223-227.
- [6] 常方圆. 基于眼动仪的智能手机 APP 图形用户界面设计可用性评估[J]. 包装工程, 2015, 36(8): 55-59.
- [7] 杨坤, 高温成, 白杰. 基于眼动指标的飞行仪表布局评估研究[J]. 人类工效学, 2016, 22(3): 1-6.
- [8] 黄升. 基于用户生活方式的智能冰箱食品管理界面设计研究[D]. 无锡: 江南大学, 2016.
- [9] 孙辛欣. 基于用户无意识行为的交互界面设计研究[D]. 无锡: 江南大学, 2013.
- [10] 孙瑞山, 陈农田. 眼动分析技术及其在航空领域的应用进展[J]. 中国民航大学学报, 2009, 27(4): 1-4, 10.
- [11] 程时伟, 石元伍, 孙守迁. 移动计算用户界面可用性评估的眼动方法[J]. 电子学报, 2009, 37(Z1): 146-150.
- [12] 孙林辉, 韩贝贝, 张伟. 基于眼动实验的英语学习类手机 APP 界面设计评价[J]. 人类工效学, 2021, 27(2): 1-8.
- [13] 刘男, 关宇涵. 基于眼动实验与层次分析法(AHP)的设计方法研究[J]. 湘潭大学自然科学学报, 2018, 40(6): 93-97.
- [14] 康丽娟. 眼动实验在设计研究中的应用误区与前景——基于国内研究现状的评述[J]. 装饰, 2017(8): 122-123.