

基于5G架构超密集组网粒子群优化算法改进*

彭 昇¹, 赵建保², 魏敏捷³

(1. 上海电力大学 电子信息工程学院, 上海 201306; 2. 国网信息通信产业集团有限公司, 北京 102200;

3. 上海电力大学 电气工程学院, 上海 201306)

摘要: 随着移动通信技术的发展, 传统智能终端设备无法满足快速增长的海量数据计算要求, 移动边缘计算为物联网中移动用户提供了低延迟和灵活的计算方案。综合考虑边缘服务器上有限的计算资源以及网络中用户的动态需求, 提出通过二进制粒子群优化算法分配发射功率优化传输能耗。将请求卸载与资源调度作为双重决策问题进行分析, 基于粒子群优化算法提出了一种新的多目标优化算法求解该问题。仿真结果表明, 二进制粒子群优化算法可以节省传输能耗, 且具有良好的收敛性。所提出的新算法在响应率方面优于现有算法, 在动态边缘计算网络中可以保持良好的性能。

关键词: 边缘计算; 资源优化; 粒子群优化; 任务卸载

中图分类号: TN929.5; TN301.6

文献标志码: A

DOI: 10.16157/j.issn.0258-7998.223278

中文引用格式: 彭昇, 赵建保, 魏敏捷. 基于5G架构超密集组网粒子群优化算法改进[J]. 电子技术应用, 2023, 49(1): 69-74.

英文引用格式: Peng Sheng, Zhao Jianbao, Wei Minjie. Improvement of particle swarm algorithm based on ultra-dense networking under 5G architecture[J]. Application of Electronic Technique, 2023, 49(1): 69-74.

Improvement of particle swarm algorithm based on ultra-dense networking under 5G architecture

Peng Sheng¹, Zhao Jianbao², Wei Minjie³

(1. College of Electronic Information Engineering, Shanghai University of Electric Power, Shanghai 201306, China;

2. State Grid Information and Telecommunication Group Co., Ltd., Beijing 102200, China;

3. College of Electrical Engineering, Shanghai University of Electric Power, Shanghai 201306, China)

Abstract: With the development of mobile communication technology, traditional intelligent terminal devices cannot meet the rapidly growing massive data computing requirements. Mobile edge computing provides low-latency and flexible computing solutions for mobile users in the Internet of Things. Considering the limited computing resources on the edge server and the dynamic needs of users in the network, this paper proposes to allocate the transmit power to optimize the transmission energy consumption through the binary particle swarm optimization algorithm. Analyzing request offloading and resource scheduling as a dual decision-making problem, a new multi-objective optimization algorithm based on particle swarm optimization algorithm is proposed to solve the problem. The simulation results show that the binary particle swarm optimization algorithm can save transmission energy and has good convergence. The proposed new algorithm outperforms existing algorithms in terms of response rate and can maintain good performance in dynamic edge computing networks.

Key words: edge computing; resource optimization; particle swarm optimization; task offloading

0 引言

随着移动通信技术的迅速发展, 物联网中的终端设备(例如智能手机、智能家居、智能汽车等)都可以通过互联网来进行相互连接^[1]。近年来, 移动设备类型及数量呈指数增长, 目前移动设备往往为了具备便携性与

易性, 而缺乏足够的计算能力及容量来满足应用的服务质量要求。移动边缘计算(Mobile Edge Computing, MEC)是物联网边端设备执行计算请求的方法^[2], 移动网络运营商与云服务提供商在边端服务器中部署丰富的计算资源, 在边端中对移动终端设备所产生的大量数据进行计算处理。

* 基金项目: 国家自然科学基金(61872230)

边缘计算资源调度的核心观点是通过优化移动边缘计算来提高计算资源与能力从而满足用户的需求。网络运营商开始普遍构建5G架构的超密集组网(Ultra-Dense Network, UDN)多基站协同服务场景^[3]。在UDN中通过部署宏基站(Macro-cell Base Station, MBS)与多个微基站(Small-cell Base Station, SBS)实现极高的频率复用,极大提高了覆盖地区的系统容量与计算能力。

粒子群优化算法(Particle Swarm Optimization, PSO)是指粒子在运动中不断更新空间内的个体最优解与全局最优解。Wang等人^[4]使用基于PSO和博弈论的MEC分配策略来最小化时间延迟;Guo等人^[5]提出一种基于遗传算法和计算资源分配算法,结果表明该算法最大限度地降低用户设备地能耗;Chen等人^[6]研究了网络中边缘服务器的数据存放问题,通过基于遗传粒子群优化算法改进降低设备数据的放置成本。本文提出二进制粒子群优化算法(Binary Particle Swarm Optimization, BPSO),通过对设备发射功率不断优化找到问题的最优解,使每台移动设备达到最小化能耗;然后,建模联合请求卸载和资源调度问题,提出一种新的启发式粒子群优化算法(JROPSO),在保证收敛的同时,提高系统的效用值。

1 问题建模

1.1 系统模型

UDN网络由一个MBS与多个SBS构成。假设当前部署的每个SBS所覆盖的是MBS的局部区域,移动终端设备U通过无线链路与SBS连接。同时MBS通过无线通信链路与多个SBS相连接来统计任务信息,计算边缘服务器属性状态和资源容量,具体的系统模型如图1所示。

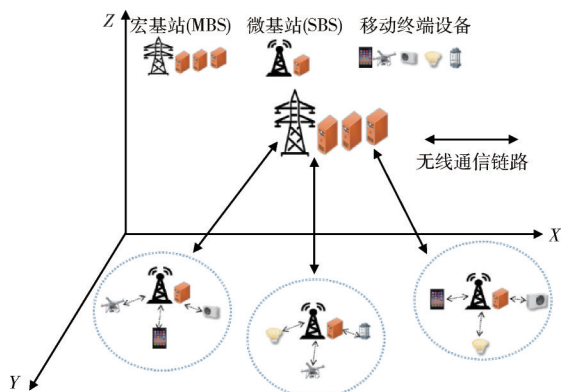


图1 系统模型

分别用 $U=1,2,\dots,u$ 和 $N=1,2,\dots,n$ 表示移动终端设备 u 和微基站 n 的集合。

1.2 通信模式

定义从移动设备 u 发射到SBS n 的发射功率 p_{un} 为 $P=\{p_{un}|u\in U, q\in Q\}$ 。移动设备 u 到SBS n 的上行链路速率 $v_{un}(t)$ 为:

$$v_{un}(t) = B \log_2 \left(1 + \frac{p_{un}(t) g_{un}(t)}{n_0 + \sum_{u' \neq u}^{U_n} p_{u'n}(t) g_{u'n}(t)} \right), u' \neq u \quad (1)$$

其中, B 表示带宽, n_0 表示高斯白噪声功率, $g_{un}(t)$ 表示信道功率增益^[7],定义为:

$$g_{un}(t) = \frac{g_0}{(x_u - x_n(t))^2 + (y_u - y_n(t))^2 + H^2}, u \in U, n \in N, t \in T \quad (2)$$

其中, g_0 表示当距离为1 m时且发射功率 $P_{un} = 1$ W时的信道功率增益, H 表示基站的高度。

1.3 计算卸载模式

完成计算任务的总能耗包括上行传输、MEC服务器上任务性能、返回数据的能耗。在本研究中,从用户角度考虑,忽略与MEC服务器上相关的能耗,只考虑上行传输能耗^[8]。这一假设是合理的,因为现有研究已经使用了这一假设^[9-12],并允许MEC服务器使用电网供电来运行。此外,与原始传输计算任务所需要的功率相比,将结果传输回设备的所需的功率较小^[13-15]。

1.4 问题制定与分析

卸载决策决定了计算任务的具体的卸载位置(MBS或SBS),本文将计算卸载决策分为两个阶段:(1)设备向基站进行任务卸载请求;(2)在SBS或MBS中执行任务。

定义 $Y = \{y_{qn}|q \in Q, n \in N\}$ 表示卸载决策矩阵,其中:

$$y_{qn} = \begin{cases} 1, & \text{基站}n\text{进行任务卸载请求} \\ 0, & \text{未进行任务卸载请求} \end{cases} \quad (3)$$

定义 $Z = \{z_{qn}|q \in Q, n \in N\}$ 表示任务请求 q 的卸载位置矩阵:

$$z_{qn} = \begin{cases} 1, & \text{任务请求}q\text{在SBS}n\text{执行} \\ 0, & \text{任务请求}q\text{在MBS执行} \end{cases} \quad (4)$$

在UDN网络中,不同移动设备的请求会卸载到微基站或者宏基站中进行处理,通过对相关值的定义来构建系统效用值。基于文献^[16],定义 q 的系统效益 k_q 。

定义系统处理请求 q 的系统代价:

$$c_q = \beta \int_{E_0 - E_t}^{E_0 - E_t + E_u^{\max}} e^{z/10} dz \quad (5)$$

其中, E_0 表示微基站的初始能量, E_t 表示微基站在时间 t 时的剩余能量, β 是使 c_q 处于 $[0,1]$ 的自定义值。

当微基站的资源容量不够处理计算请求时,移动设备将会发送请求至宏基站,所以卸载到宏基站的额外成本为:

$$e_q = \eta k_q + (1 - \eta) E_q^{\text{pro}} \quad (6)$$

其中, η 是一个自定义常数, 表示系统效益与能耗的相对性。由上述定义, 系统总效用值可以定义为:

$$W = \sum_n \sum_q \left[z_{qn} (k_q - c_q) - (1 - z_{qn}) e_q \right] \quad (7)$$

2 BPSO 和 JROPSO 算法

本节首先描述了发射功率分配问题, 证明发射功率与能耗的函数是一个拟凸函数, 并提出标准二进制粒子群优化算法(Binary Particle Swarm Optimization, BPSO)。针对资源调度与请求卸载问题, 提出一种基于 PSO 的算法联合优化资源分配和请求卸载问题(PSO based Joint Request Offloading, JROPSO)。

2.1 BPSO 算法

等式(1)、(2)可以具体表示为:

$$p_{un}(t)T = \frac{p_{un}(t)I_u}{v_{un}(t)} = \max_p E = \sum_n \sum_u \frac{p_{un}(t)I_u}{B \log_2 \left(1 + \frac{p_{un}(t)g_{un}(t)}{n_0 + \sum_{u'} p_{u'n}(t)g_{u'n}(t)} \right)} \quad (8)$$

约束条件为: $0 \leq p_{un}(t) \leq p_{\max}, \forall u \in U, n \in N$ 。其中 I_u 表示终端设备 u 的任务量。假设一个特定的区域, 等式(9)表示为 p_{un} 与 E 的关系:

$$\min E = \sum_u \frac{p_{un} I_u}{B \log_2 (1 + \delta p_{un})} \quad (9)$$

粒子位置 $X_k = \{X_{k1}, X_{k2}, \dots, X_{km}, \dots, X_{kM}\}$ 表示所有任务卸载到相应 MEC 服务器。粒子速度 $V_k = \{V_{k1}, V_{k2}, \dots, V_{kn}, \dots, V_{kM}\}$ 表示任务卸载到其他服务器的时间跨度。其中, k 表示 k 个粒子的群, m 是移动设备总数。

$$V_k = \omega V_k + c_1 r_1 (L_k - X_k) + c_2 r_2 (G - X_k) \quad (10)$$

$$X_k = X_k + V_k \quad (11)$$

其中, L_k 代表粒子的最佳单个位置, G 代表粒子的最佳全局位置, ω 为恒定惯性重量, c_1 、 c_2 表示两个加速度系数, r_1 、 r_2 表示(0, 1)中的随机变量。

通过使用 sigmoid 函数将速度转换为 [0, 1] 内的概率:

$$\text{Sig}(V_{km}) = \frac{1}{1 + e^{-V_{km}}} \quad (12)$$

该粒子位置根据以下规则转换为二进制变量:

$$X_{km} = \begin{cases} 0, \text{rand} < \text{Sig}(V_{km}) \\ 1, \text{rand} \geq \text{Sig}(V_{km}) \end{cases} \quad (13)$$

通过定义自适应度函数来评估解的可行性。Fit(X_k) 表示每个粒子在 X_k 处的自适应度, 定义自适应度函数为:

$$\text{Fit} = \sum_{u=1}^{U_n} \left(\frac{p_{un} I_u}{B \log_2 (1 + \delta p_{un})} \right) + P \quad (14)$$

其中, $P = \sum_{n=1}^N \theta_n \left(\max \left(0, \sum_{u=1}^U y_{un} - M \right) \right)$, θ_n 是惩罚函数, 帮助粒子尽快找到可行区域以便更新全局位置。BPSO 算法如下。

算法 1: 标准二进制粒子群优化算法(BPSO)

(1) 随机生成粒子数 k 的速度 v_k 和位置 x_k

(2) repeat

(3) for 每个 $k \in K$ do

(4) 计算适应度值

(5) if Fit(X_k) > Fit(L_k) then

(6) 更新当前单个最优解 L_k

(7) end

(8) if Fit(L_k) > Fit(G) then

(9) 更新当前全局最优解 G

(10) end

(11) 根据式(10)、式(11)更新粒子速度变量

(12) 在[0, 1]中生成均匀随机数 rand

(13) 根据式(13)更新二进制粒子位置变量

(14) end

(15) until 达到最大迭代次数或者约束条件

2.2 JROPSO 算法

所以以上可以分为两个问题: 请求卸载问题与资源调度问题。

请求卸载问题:

$$\max_Z M = \sum_n \sum_q \left((1 - z_{qn}) (\eta k_q + (1 - \eta) E_q^{\text{pro}}) \right) \quad (15)$$

资源调度问题:

$$\max_{Z, Y} W = \sum_n \sum_q \left[z_{qn} (k_q - c_q) - (1 - z_{qn}) (\eta k_q + (1 - \eta) E_q^{\text{pro}}) \right] \quad (16)$$

将粒子群规模 K 设置为 25, 迭代次数设置为 5000 次, 设置参数 $\omega = 2, c_1 = c_2 = 2$ 。JROPSO 算法如下。

算法 2: 基于粒子群优化算法的联合资源分配和计算卸载决策算法(JROPSO)

输入: 粒子群大小 K , 最大迭代次数 T_{\max} , 收敛标准 ζ , 参数 ω 、 c_1 、 c_2 ;

输出: 全局最优个体。

(1) 初始化: 随机位置 X_k 与随机速度 V_k 的粒子群 K , 迭代指数设置为 $t=1$ 。

(2) 根据式(14)计算每个粒子的适应度值

(3) repeat:

(4) for 每个粒子数 K_m do

(5) 分别根据式(10)、式(11)更新速度 V_k 和位置 X_k

- (6) end
- (7) 根据等式(14)计算自适应度
- (8) 进行比较,更新当前单个最优解
- (9) 更新 K 个粒子中的全局最优解
- (10) 增加迭代指数 $t=t+1$
- (11) until $t > T_{\max}$ 或 $\text{Fit}(t) - \text{Fit}(t-1) \leq \zeta$
- (12) 输出全局最优个体

3 实验

3.1 实验介绍与参数设计

为了评估所提出算法的有效性,本文在 Matlab R2019-b 中进行仿真实验。本次仿真场景由多个移动设备、一个宏基站、多个微基站构成,根据移动设备所在的位置确定进行边缘计算的区域,与其对应的微基站进行关联。具体仿真参数见表1。

表1 仿真参数

参数	设置值
u	{12,20,32,40,60}
n	{3,5,8,10,13,18}
C_q/MHz	1000~2000
D_q/KB	600~1000
T_{g_q}/s	[0.4,0.6]
T_{b_q}/s	$T_{g_q}+[0.1,0.15]$
f_n^{\max}/GHz	60
f_c^{\max}/GHz	120
B/MHz	20
H/m	10
n_0/dBm	-100
p_{\max}/W	{4,5,6}

本文将BPSO和JROPSO算法性能与以下具有代表性的方法进行比较:模拟退火算法(SAPA)^[17]、基于多区域随机调度请求卸载策略算法(ROGS)^[18]和基于二分法的调度优化策略改进算法(HOBS)^[19]。

3.2 实验结果

3.2.1 BPSO算法

收敛曲线展示两种算法迭代结果后的收敛情况,绘制得到的两种算法的收敛曲线如图2所示。

BPSO算法与经典模拟退火(SAPA)算法比较表明,BPSO算法不仅能耗优于SAPA算法,同时BPSO的收敛效果也较好,可以在短时间内达到最小能耗。

在 $U = \{12, 20, 32, 40, 60\}$ 且 $p_{\max} = 5 \text{ W}$ 的情况下不同算法的能耗值对比图如图3所示。设定不同的 p_{\max} 参数,BPSO的能耗如图4所示。

从图3可知,总体能耗随着移动设备数量增加而提高,BPSO的能耗在不同移动设备数量情况下始终小于SAPA,说明BPSO起到了更好的优化作用。从图4可

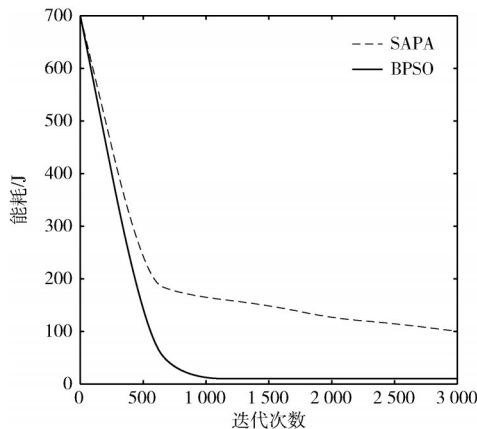


图2 算法能耗值优化收敛曲线

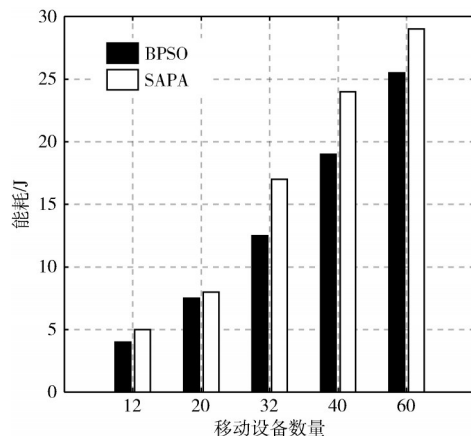


图3 能耗与移动设备数量关系图

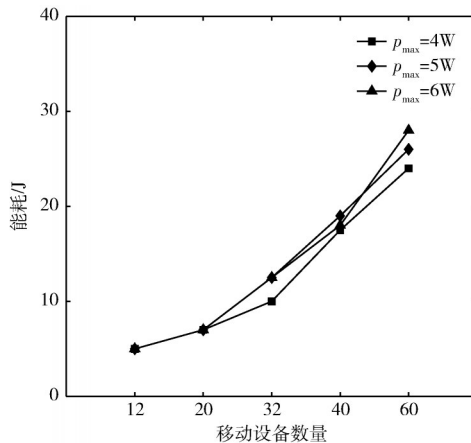


图4 能耗与最大功率关系图

知, p_{\max} 越大, 能耗越大。因为系统设定最大功率越大, 设备传输的平均传输功率增加, 所造成系统能耗增加。

3.2.2 JROPSO性能评估

假设系统中所有微基站的计算能力相同, $f_n = 70 \text{ GHz}$, 并且移动设备的卸载请求是相同的, $C_q = 1500 \text{ MHz}$, $D_q = 700 \text{ KB}$, $T_{g_q} = 0.5 \text{ s}$, $T_{b_q} = 0.65 \text{ s}$ 。不同工作负载在移动设备数量 $U = \{12, 20, 32, 40, 60\}$ 时的效用值曲线如图5所示。

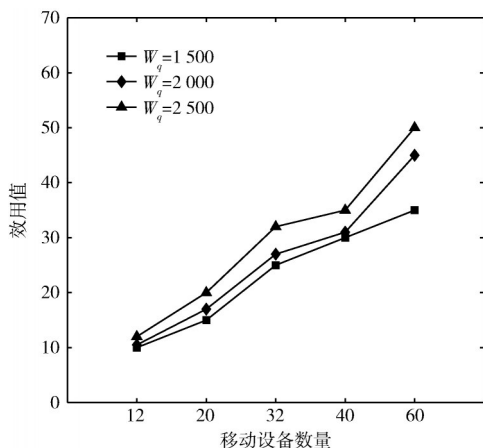


图5 不同工作负载与系统效用值关系

从图5可知,系统效用值随着设备增多而增加。当设备数量一定时,工作负载增加,效用值降低。因为系统中计算资源有限,当工作负载增加会占用计算资源,而导致效用值降低。

据工作负载的输入大小来比较JROPSO的性能,可以得到不同 C_q 值下的系统效用值,绘制得到的关系图如图6所示。

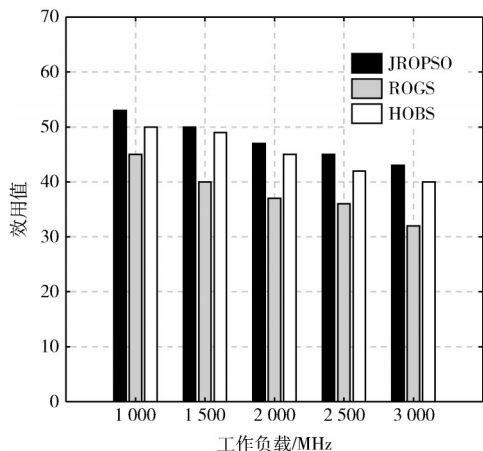


图6 不同工作负载下的效用值

由图6可知,在($U=60, D_q=700\text{ KB}$)不同工作负载的情况下,JROPSO的性能总是优于其他算法的性能。

定义系统的响应率为请求容忍延迟内完成计算数量与请求总数的比值,由于微基站计算能力相同,因此假设微基站中的平均功率 P_{SBS} 相同,因此可以转换为实际消耗能耗 E_u^{pro} 与总消耗能耗的比值 E_q^{pro} 。在 $U=60, D_q=700\text{ KB}$ 的情况下,不同工作负载产生系统响应率曲线如图7所示。

由图7可知,JROPSO响应率曲线高于另外两种算法。随着工作负载的增加,系统效用值降低,响应率减少,但是随着工作负载的增加,JPOPSO响应率下降速度加快,而HOBS响应率下降速度保持稳定,可见HOBS在高工作负载情况下表现较好。

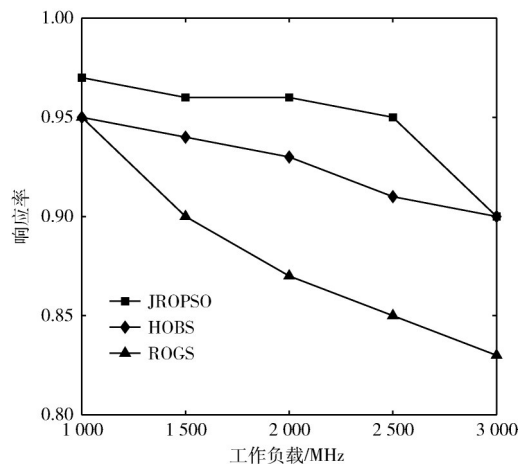


图7 不同工作负载与响应率关系

不同算法分别在不同请求数据下得到的效用值关系与响应率曲线如图8和图9所示。

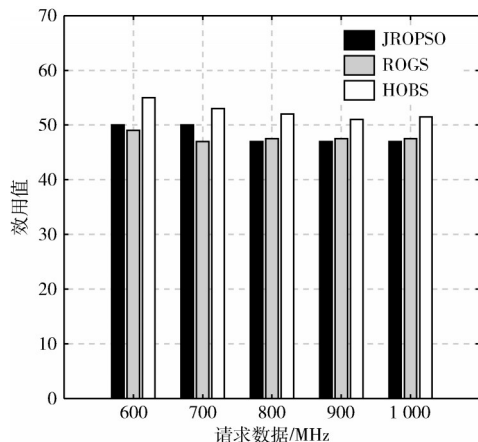


图8 不同请求数据下的效用值

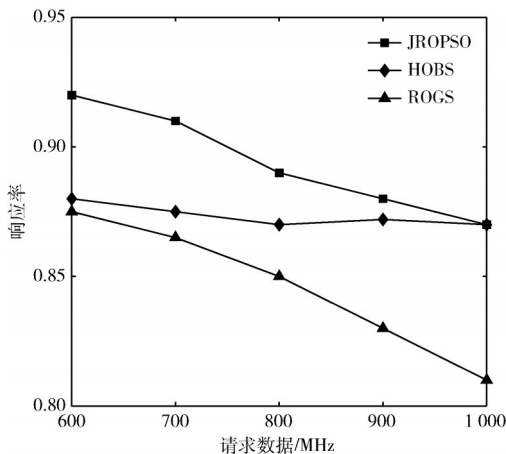


图9 不同请求数据下的响应率

JROPSO在效用值方面略优于ROGS,但是不及HOBS。JROPSO的响应率表现最好,在优化过程中降低了系统效率来提高系统响应率。并且随着请求负载的增加,大量数据

的输入增加了系统的传输延迟,导致系统响应率的降低。

4 结论

边缘计算已经成为5G网络中的前沿技术,边缘计算能够将计算任务从移动设备卸载到边缘服务器,来解决特定于移动设备的限制。本文开发了一种高效算法,用于5G网络中多边缘服务器体系结构的计算卸载策略。本文提出BPSO算法,通过优化发射功率使系统传输能耗最小化;然后研究资源分配与计算卸载决策策略,开发了JROPSO算法,得到了在不同请求数据和请求负载下相较ROGS、HOBOS算法更好的效果。未来拟对于资源分配与计算卸载双重决策问题进一步深入研究,寻求更优化的解决方案。

参考文献

- [1] ELGENDY I A, ZHANG W Z, ZENG Y, et al. Efficient and secure multi-user multi-task computation offloading for mobile-edge computing in mobile IoT networks[J]. IEEE Transactions on Network and Service Management, 2020, 17(4): 2410-2422.
- [2] 施巍松,张星洲,王一帆,等.边缘计算:现状与展望[J]. 计算机研究与发展,2019,56(1):69-89.
- [3] GE X, TU S, MAO G, WANG G X, et al. 5G ultra-dense cellular networks[J]. IEEE Wireless Commun., 2016, 23(1): 72-79.
- [4] WANG Y, MIN S, WANG X, et al. Mobile-edge computing:partial computation offloading using dynamic voltage scaling[J]. IEEE Transactions on Communications, 2016, 64(10): 4268-4282.
- [5] GUO F, ZHANG H, JI H, et al. Energy efficient computation offloading for multi-access MEC enabled small cell networks[C]//2018 IEEE International Conference on Communications Workshops. IEEE, 2018:1-6.
- [6] CHEN X, ZHANG J, LIN B, et al. Energy-efficient offloading for DNN-based smart IoT systems in cloud-edge environments[J]. IEEE Transactions on Parallel and Distributed Systems, 2022, 33(3): 683-697.
- [7] JEONG S, SIMEONE O, KANG J. Mobile edge computing via a UAV-mounted cloudlet: optimization of bit allocation and path planning[J]. IEEE Transactions on Vehicular Technology, 2017,67(3):2049-2063.
- [8] YOU C, HUANG K, CHAE H, et al. Energy-efficient resource allocation for mobile-edge computation offloading[J]. IEEE Transactions on Wireless Communications, 2017, 16(3):1397-1411.
- [9] ZHANG J, HU X, NING Z, et al. Energy-latency trade-off for energy-aware offloading in mobile edge computing networks[C]//2018 IEEE Internet Things. IEEE, 2018:2633-2645.
- [10] BI J, YUAN H, DUANMU S, et al. Energy-optimized partial computation offloading in mobile-edge computing with genetic simulated-annealing-based particle swarm optimization[J]. IEEE Internet of Things Journal, 2021, 8(5):3774-3785.
- [11] CHEN M, HAO Y. Task offloading for mobile edge computing in software defined ultra-dense network[J]. IEEE Areas Commun, 2018, 36(2):587-597.
- [12] GUO H, LIU J. Collaborative computation offloading for multiaccess edge computing over fiber-wireless networks[J]. IEEE Communications Magazine, 2018, 56(8): 14-19.
- [13] 张海波,李虎,陈善学,等.超密集网络中基于移动边缘计算的任务卸载和资源优化[J].电子与信息学报, 2019, 41(5): 1194-1201.
- [14] DENG X, SUN Z, LI D, et al. User-centric computation offloading for edge computing[J]. IEEE Internet of Things Journal, 2021, 8(16): 12559-12568.
- [15] CHOJ J, AHN S. Scalable service placement in the fog computing environment for the IoT-based smart city[J]. Journal of Information Processing Systems, 2019, 15(2): 440-448.
- [16] NAKAMURA Y, MIZUMOTO T, SUWA H, et al. In-situ resource provisioning with adaptive scale-out for regional IoT services[C]//2018 IEEE/ACM Symposium on Edge Computing(SEC). IEEE,2018:203-213.
- [17] WU D, YAN J, WANG H, et al. User-centric edge sharing mechanism in software-defined ultra-dense networks[J]. IEEE Journal on Selected Areas in Communications, 2020, 38(7): 1531-1541.
- [18] SARDELLITTI S, CUTARI G, BARBAROSSA S. Joint optimization of radio and computational resources for multicell mobile-edge computing[J]. IEEE Trans. Signal Inf. Process. Netw., 2015,1(2):89-103.
- [19] TRAN T X, POMPILO D. Joint task offloading and resource allocation for multi-server mobile-edge computing networks[J]. IEEE Transactions on Vehicular Technology, 2018,68(1): 856-868.

(收稿日期:2022-08-20)

作者简介:

彭昇(1998-),男,硕士,主要研究方向:通信工程、电力物联网。

赵建保(1968-),男,博士,教授,主要研究方向:电力信息化、信息与通信工程。

魏敏捷(1982-),通信作者,男,博士,助理研究员,主要研究方向:电力信息技术、人工智能, E-mail: weiminjie@shiep.edu.com。



扫码下载电子文档

版权声明

经作者授权，本论文版权和信息网络传播权归属于《电子技术应用》杂志，凡未经本刊书面同意任何机构、组织和个人不得擅自复印、汇编、翻译和进行信息网络传播。未经本刊书面同意，禁止一切互联网论文资源平台非法上传、收录本论文。

截至目前，本论文已经授权被中国期刊全文数据库（CNKI）、万方数据知识服务平台、中文科技期刊数据库（维普网）、DOAJ、美国《乌利希期刊指南》、JST 日本科技技术振兴机构数据库等数据库全文收录。

对于违反上述禁止行为并违法使用本论文的机构、组织和个人，本刊将采取一切必要法律行动来维护正当权益。

特此声明！

《电子技术应用》编辑部

中国电子信息产业集团有限公司第六研究所