

DOI:10.19651/j.cnki.emt.2519784

能耗约束下量子遗传算法时延最小化卸载策略^{*}郭澳归¹ 叶成荫¹ 申佳欣²

(1. 辽宁石油化工大学信息与控制工程学院 抚顺 113001; 2. 西安邮电大学通信与信息工程学院 西安 710121)

摘 要: 针对移动边缘计算中时延最小化的任务卸载问题,提出了一种在能耗约束下实现计算任务总时延最小化的量子遗传算法。首先,以最小总时延的卸载策略为优化目标,建立了系统模型和计算模型;其次,权衡总时延最小化与能耗、计算资源的约束的条件,构建了总时延最小化的目标函数;最后,将量子遗传算法与量子变异门、精英保留策略相结合得到总时延最小化的卸载策略。仿真实验结果显示,与其他卸载策略相比,所提出的卸载策略在不同的能耗约束、计算任务数下总时延更小,且在不同任务数据量下,其总时延相比遗传算法减少了7.3%、相比自适应粒子群优化算法减少了4.3%。

关键词: 移动边缘计算;任务卸载;量子遗传算法;卸载策略;能耗约束

中图分类号: TN929.5 **文献标识码:** A **国家标准学科分类代码:** 510

Energy consumption constraint-based quantum genetic algorithm
delay minimization offloading strategyGuo Aogui¹ Ye Chengyin¹ Shen Jiaxin²

(1. School of Information and Control Engineering, Liaoning Petrochemical University, Fushun 113001, China;

2. School of Communications and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710121, China)

Abstract: For the task offloading problem of minimizing latency in mobile edge computing, a quantum genetic algorithm is proposed to achieve the minimum total latency of computing tasks under energy consumption constraints. Firstly, with the offloading strategy minimizing the total latency as the optimization objective, a system model and a computing model are established; secondly, by balancing the conditions of minimizing total latency with energy consumption and computing resources constraints, a target function for minimizing total latency is constructed; finally, by combining the quantum genetic algorithm with quantum mutation gates and elite retention strategies, an offloading strategy for minimizing total latency is obtained. Simulation experiments show that compared with other offloading strategies, the proposed offloading strategy has a smaller total latency under different energy consumption constraints and different numbers of computing tasks, and its total latency is reduced by 7.3% compared with the genetic algorithm and by 4.3% compared with the adaptive particle swarm optimization algorithm under different task data volumes.

Keywords: mobile edge computing; task offloading; quantum genetic algorithm; offloading strategy; energy consumption constraint

0 引言

在移动边缘计算(mobile edge computing, MEC)架构中,任务卸载是核心技术,能依据任务量、时延及能耗需求决定任务执行位置,对提升移动设备服务质量与续航至关重要^[1]。随着智能终端数量激增,在类似校园生活区、密集居民区、写字楼等单基站覆盖区域的单小区多用户场景中,

多用户常发起视频渲染等计算任务,却受单基站资源范围限制^[2],面临移动设备计算资源有限的问题,全本地执行易因时延过大,影响体验且耗电快^[3];单小区 MEC 服务器虽算力强,多任务同时卸载容易导致其资源过载,设备上传任务的传输能耗会超限^[4]。可见,该场景是 MEC 落地高频场景,针对其优化卸载策略,能直接解决实际环境用户需求。

收稿日期:2025-09-04

^{*} 基金项目:兴辽人才计划项目(XLYC2203160)资助

针对 MEC 任务卸载与资源分配优化,学者们开展了大量研究,但成果存在不足。在单目标优化研究中,吴波等^[5]的改进粒子群算法、Li 等^[6]的并行协作遗传算法(genetic algorithm,GA)、王丽等^[7]的 GA 均聚焦时延最小化,却未充分考虑 MEC 服务器计算资源、设备能耗或本地计算资源约束;丁嘉伟^[8]的多变量联合优化算法、Zhuo 等^[9]的可重构智能表面辅助无人机 MEC 系统,仅以能耗最小化为目标,无法满足时延需求。

部分多目标研究也未覆盖约束体系。Eang 等^[10]的拉格朗日对偶优化算法虽兼顾总成本与总延迟,却未考虑本地及 MEC 服务器资源限制;武晓雯等^[11]的改进遗传粒子群算法解决了传统算法易陷入局部最优、寻优能力不足的问题,通过解决无人机路径寻优的问题,为解决寻优问题提供了新思路;顾敏等^[12]的自适应粒子群优化算法(adaptive particle swarm optimization,APSO)算法在单小区多用户 MIMO-MEC 网络中最小化总时延,但未考虑 MEC 服务器计算资源上限;辜航等^[13]的 GA-深度学习方案以最小化系统总成本为目标,但并未考虑本地计算资源上限;赵婵婵等^[14]的元强化学习方法虽能优化异构任务的时延或能耗,却未建立“能耗-本地资源-MEC 资源”联合约束模型。

在单小区多用户场景中,实时在线实验、远程医疗会诊等核心任务,时延过高会直接导致服务中断,而能耗是移动设备基础运行约束。在能耗不超设备上限前提下最小化总时延,本质是优先解决用户体验与能耗约束、计算资源约束的矛盾。

综上,现有研究或聚焦单目标,或多目标研究忽略关键资源约束,尤其在单小区多用户这一高频场景下,缺乏同时满足能耗、本地计算资源与 MEC 服务器计算资源约束的总时延最小化卸载策略。为此,本文针对该场景,提出融合量子变异门与精英保留策略的量子遗传算法(quantum genetic algorithm, QGA),通过构建多约束系统及计算模型、设计目标函数,高效获取最优卸载策略,并通过仿真验证其优越性。

1 系统模型

建立如图 1 所示的单小区多用户的 MEC 系统模型。

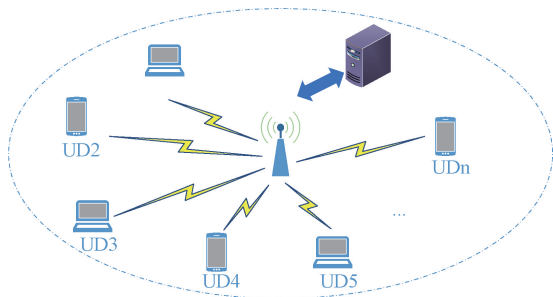


图 1 系统模型
Fig. 1 System model

其中包含一个 MEC 服务器,一个基站(base station, BS)和 n 个移动设备(user device, UD),这个小区内的每一个 UD 都可以通过无线链路连接到该小区的 BS 上,然后通过 BS 将任务上传给 MEC 服务器来处理任务。在该场景下,下行链路的传输速率将远远高于上行链路传输速率,且由 MEC 服务器处理后的数据量也将远远小于上传的数据量,故本研究的下行链路的传输时延可以忽略不计。

假设第 i 个移动设备 UD_i 仅产生一个计算任务 $Task_i$, $i \in N, N = \{1, 2, \dots, n\}$, 且 $Task_i = \{B_i, U_i, E_i^{\max}\}$, 其中 B_i 为该任务卸载到 MEC 服务器的数据量, $U_i = u_i B_i$ 为处理该任务消耗的 CPU 周期数, $E_i^{\max}$ 为处理该任务时设备所能接受的最大的能耗。对于 n 个计算任务组成的卸载策略的集合为 $\mathbf{S} = \{s_1, s_2, \dots, s_n\}$, s_i 为计算任务 $Task_i$ 的分配方式且 $s_i \in \{0, 1\}$, $i \in N$ 。如果 $s_i = 0$ 表示计算任务 $Task_i$ 在本地执行, $s_i = 1$ 表示计算任务 $Task_i$ 卸载到边缘服务器上执行。

用户将计算任务 $Task_i$ 的数据上传到该小区基站的上行链路的传输速率 R_i 定义为:

$$R_i = W \log_2 \left(1 + \frac{P_i H_i}{N_0} \right) \quad (1)$$

式中: W 为信道带宽; P_i 为移动设备 UD_i 的发射功率; H_i 为移动设备 UD_i 与 MEC 服务器之间的信道增益; N_0 为信道高斯噪声功率。

2 问题建模

2.1 计算时延

1) 本地计算时延

计算任务 $Task_i$ 在本地执行的时延 T_i^{local} 取决于任务的计算量 U_i 和移动设备 UD_i 的计算资源, 定义为:

$$T_i^{local} = \frac{U_i}{f_i^{local}} \quad (2)$$

式中: f_i^{local} 为移动设备分配给任务 $Task_i$ 的计算资源。

2) 边缘卸载时延

由于本地设备存储量和计算能力的限制,处理一些计算量大且存储需求量较高的任务时存在困难,这时需将任务卸载到 MEC 服务器上执行。

计算任务 $Task_i$ 卸载到边缘服务器执行时所消耗的时间包括传输阶段的时延 $T_{i,trans}^{mec}$ 、处理阶段的时延 $T_{i,com}^{mec}$ 和回程阶段的时延 $T_{i,return}^{mec}$, 但考虑到 $T_{i,return}^{mec}$ 远小于 $T_{i,trans}^{mec}$, 所以 $T_{i,return}^{mec}$ 忽略不计。

计算任务 $Task_i$ 在处理阶段的时延 $T_{i,com}^{mec}$ 定义为:

$$T_{i,com}^{mec} = \frac{U_i}{f_i^{mec}} \quad (3)$$

式中: f_i^{mec} 为 MEC 服务器为任务 $Task_i$ 分配的计算资源。

计算任务 $Task_i$ 在传输阶段的时延 $T_{i,trans}^{mec}$ 定义为:

$$T_{i,trans}^{mec} = \frac{B_i}{R_i} \quad (4)$$

由式(3)和(4)可知计算任务 $Task_i$ 在 MEC 服务器执行的时延 T_i^{mec} 为:

$$T_i^{mec} = T_{i,com}^{mec} + T_{i,trans}^{mec} = \frac{U_i}{f_i^{mec}} + \frac{B_i}{R_i} \quad (5)$$

3) 总时延

由式(2)和(5)可知计算任务 $Task_i$ 的总时延 T_i^{total} 为:

$$T_i^{total} = s_i T_i^{mec} + (1 - s_i) T_i^{local} \quad (6)$$

2.2 计算能耗

1) 本地计算能耗

计算任务 $Task_i$ 在本地执行的能耗 E_i^{local} 定义为:

$$E_i^{local} = k (f_i^{local})^2 U_i \quad (7)$$

式中: k 为设备芯片的相关参数^[15]。

2) 边缘卸载能耗

计算任务 $Task_i$ 在传输阶段的能耗 $E_{i,trans}^{mec}$ 定义为:

$$E_{i,trans}^{mec} = P_i T_{i,trans}^{mec} = P_i \frac{B_i}{R_i} \quad (8)$$

计算任务 $Task_i$ 在处理阶段(此时移动设备处于空闲状态)的能耗 $E_{i,com}^{mec}$ 定义为:

$$E_{i,com}^{mec} = P_i^f T_{i,com}^{mec} = P_i^f \frac{U_i}{f_i^{mec}} \quad (9)$$

式中: P_i^f 为移动设备 UD_i 处于空闲时功率。

由式(8)和(9)可知计算任务 $Task_i$ 卸载到 MEC 服务器执行的能耗 E_i^{mec} 为:

$$E_i^{mec} = E_{i,com}^{mec} + E_{i,trans}^{mec} = P_i^f \frac{U_i}{f_i^{mec}} + P_i \frac{B_i}{R_i} \quad (10)$$

3) 总能耗

由式(7)和(10)可知计算任务 $Task_i$ 的总能耗 E_i^{total} 为:

$$E_i^{total} = s_i E_i^{mec} + (1 - s_i) E_i^{local} \quad (11)$$

2.3 总时延最小化模型

在边缘计算场景中,任务执行过程易产生较高能耗,而这会导致设备处理效率下降。为此,可将能耗、本地计算资源及 MEC 服务器计算资源作为约束条件,通过限制每个计算任务的能耗与所使用的计算资源,进而得到总时延最小化的卸载策略。结合式(6)、(7)和(10)建立目标优化函数,公式为:

$$\begin{cases} \min T^{total} = \sum_{i=1}^n T_i^{total}(s_i) \\ s.t. \\ C1: E_i^{total} < E_i^{\max}, i \in N \\ C2: 0 \leq \sum_{i=1}^n s_i f_i^{mec} \leq f_{\max}^{mec}, i \in N \\ C3: 0 \leq f_i^{local} \leq f_{\max}^{local}, i \in N \\ C4: s_i \in \{0, 1\}, i \in N \end{cases} \quad (12)$$

式中: $f_{\max}^{mec}$ 为 MEC 服务器的最大计算资源; $f_{\max}^{local}$ 为任务对应的设备最大的计算资源。C1 表示执行计算任务 $Task_i$ 的最大值不超过设备能接受的最大损耗; C2 表示保证所有在 MEC 服务器上执行任务所分配的計算资源之和不超过

该服务器所承受的最大计算资源; C3 表示计算任务对应的移动设备在处理任务时所获得计算资源不能超过该设备最大的计算资源; C4 表示计算任务选择的卸载策略。

3 基于 QGA 的卸载策略

QGA 是一种融合量子计算与 GA 的算法,它既具备 GA 的全局搜索能力,又借助了量子计算的并行性及量子态的叠加特性,因此在解决复杂优化问题时展现出更高的效率与性能。然而,传统量子遗传算法(traditional quantum genetic algorithm, TQGA)存在染色体多样性有限的问题,为此本研究在 QGA 中引入量子变异门与精英保留策略操作,使算法能够通过对目标函数进行寻优,更快地得到移动边缘计算中任务分配的最优策略。其中 QGA 寻优的过程与任务卸载策略的关系如表 1 所示。

表 1 QGA 寻优过程与任务卸载策略的关系

Table 1 The relationship between the QGA optimization process and the task offloading strategy

QGA 寻优的过程	任务卸载策略
量子比特	计算任务的分配方式
染色体中的基因数	计算任务的数量
量子染色体的种群	卸载方案的集合
适应度最优的染色体	任务卸载的最佳策略
最优适应度值	最小总时延

3.1 量子编码及初始化

在 QGA 中,引入量子比特来实现基因的存储和表达。一个量子染色体表示一个卸载策略,量子染色体中的一个量子比特(qubit)表示一个计算任务对应的卸载方式,可与经典位的 0 和 1 进行对应编码,量子比特可用 2 个量子态的本征态 $|0\rangle$ 和 $|1\rangle$ 来表示。染色体的基因由两个量子比特的状态来表示。其中,一个量子比特不仅能呈现“0”或“1”态,还能表现这两种状态的任意叠加态,其形式为一种线性组合。这样该染色体便包含了所求问题的所有可能信息,而非仅代表某一个确定的信息,一个量子比特编码可以描述为:

$$|\varphi\rangle = \alpha |0\rangle + \beta |1\rangle \quad (13)$$

式中: α 和 β 为量子概率幅,表示对应的量子比特,是一对复数。量子比特处于 $|0\rangle$ 态的概率表示为 $|\alpha|^2$, 量子比特处于 $|1\rangle$ 态的概率表示为 $|\beta|^2$, 且满足归一化,即 $|\alpha|^2 + |\beta|^2 = 1$ 。

将卸载策略作为算法中种群的个体,第 t 代包含 m 个个体的量子种群表示为 $Q(t) = \{q_1^t, q_2^t, \dots, q_m^t\}$, 其中 q_j^t 为第 t 代第 j 个量子染色体,表示形式为:

$$q_j^t = \begin{bmatrix} \alpha_1^t & \alpha_2^t & \dots & \alpha_n^t \\ \beta_1^t & \beta_2^t & \dots & \beta_n^t \end{bmatrix}, j \in M \quad (14)$$

式中: n 为量子染色体中的量子比特数, $M = \{1, 2, \dots, m\}$

为种群的集合。

在 QGA 开始时,随机生成以 n 个量子比特为编码的染色体,对种群中个体的全部量子比特进行初始化,即所有量子染色体 q_i^t 中的概率幅 α_i^t 和 β_i^t 都等于 $1/\sqrt{2}$, $i \in N$, 并按照种群大小生成初始化种群 $Q(t_0)$ 。

3.2 量子测量

在测量过程中,将量子比特转换为二进制比特。测量步骤包括:

- 1) 根据量子概率幅的平方生成随机数 $r \sim U(0,1)$ 。
- 2) 当随机数 r 小于第 i 个量子比特处于 $|0\rangle$ 态的概率 $|\alpha_i|^2$ 时,基因坍塌为 0;否则,基因坍塌为 1。
- 3) 量子比特测量后坍塌为二进制比特,用于后续的适应度评估。公式为:

$$s_i = \begin{cases} 0, & r \leq |\alpha_i|^2 \\ 1, & \text{其他} \end{cases} \quad (15)$$

式中: s_i 为第 i 个量子比特坍塌为二进制比特的值。

根据上述步骤,经过测量量子种群 $Q(t)$ 得到二进制种群 $P(t) = \{S_1^t, S_2^t, \dots, S_m^t\}$, 其中 S_j^t 为第 t 代第 j 个二进制染色体,表示形式为:

$$S_j^t = [s_1, s_2, \dots, s_n], j \in M \quad (16)$$

3.3 适应度函数设计

为评估卸载策略的优劣,在 QGA 中需为所优化的问题设计合适的适应度函数,通过该函数计算卸载策略的适应度值,进而判断卸载策略的优劣。为了得到满足能耗限制、计算资源约束且总时延最小的最优卸载策略,将能耗约束、本地计算资源和 MEC 服务器计算资源作为惩罚项引入,得到的适应度函数为:

$$f = \frac{1}{T^{total} + \lambda \cdot (pen_1 + pen_2 + pen_3)} \quad (17)$$

其中, λ 为惩罚因子且 $\lambda \in (0,1)$, $pen_1 = \max(0,$

$$E_i^{total} - E_i^{max}), pen_2 = \max(0, \sum_{i=1}^n S_i f_i^{mec} - f_{max}^{mec}), pen_3 = \max(0, f_i^{local} - f_{max}^{local})$$

该函数通过惩罚项的机制将多约束优化问题转化为单目标函数,既精确量化了总时延性能,又通过 λ 权重平衡了时间效率与约束满足的权衡,其中倒数转换将最小化问题转为最大化问题,这样的适应度函数更符合 QGA 的选择机制,从而有效引导算法搜索同时满足能耗限制、计算资源约束且总时延最小的最优卸载策略。

3.4 精英保留策略的设计

QGA 引入精英保留策略的优点是在可在确保精英个体不会丢失的同时,引导种群向更优方向进化。这样既保证每一代的最佳适应度不会变差,也保证了算法的单调收敛性。其步骤包括精英数量的计算、精英选择、精英保留等部分,步骤包括:

- 1) 精英数量的计算:依据种群规模和精英保留比例 E_r , 确定需保留的精英个体数量,确保至少保留 1 个最优个体,

计算公式为:

$$E_{size} = \max(1, m \cdot E_r) \quad (18)$$

- 2) 精英选择:根据精英个体的数量在第 1 代之后的每代进化中,从当前种群中筛选出适应度最高的前 E_{size} 个个体作为精英。

- 3) 精英保留:在计算新一代种群的适应度值,找出下一代中适应度最差的个体,将选定的精英个体替换下一代中适应度最差的个体。

3.5 量子旋转门

量子旋转门通过调整量子比特的概率幅,能够直接引导种群向最优解方向进化,相比传统 GA 的交叉和变异操作,具有更快的收敛速度。根据当前种群的最优解,使用量子旋转门策略更新量子概率幅。如果当前解比最优解差,则根据最优解的决策方向调整量子概率幅。量子旋转门的矩阵表示为:

$$R(\theta_i) = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{bmatrix} \quad (19)$$

式中: θ_i 为旋转角度,角度的大小根据一个事先设计的调整策略确定。

对于当前染色体的第 i 个基因的量子叠加态 $|\varphi_i\rangle = \alpha_i|0\rangle + \beta_i|1\rangle$ 进行量子旋转门操作,即:

$$\begin{bmatrix} \alpha'_i \\ \beta'_i \end{bmatrix} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{bmatrix} \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} \quad (20)$$

式中: $[\alpha_i \ \beta_i]^T$ 为当前染色体的第 i 个基因, $[\alpha'_i \ \beta'_i]^T$ 为当前染色体的第 i 个基因经过量子旋转门更新后的新基因。

更新当前个体 x , 需将当前个体 x 的适应度值 $f(x)$ 与最优个体 $best$ 的适应度值 $f(best)$ 比较确定当前个体 x 的更新方向,如果 $f(best) > f(x)$, 改变 x 中对应的量子比特,将概率幅向最优个体的方向调整;反之,如果 $f(best) < f(x)$, 改变 x 中对应的量子比特,将概率幅向最优个体的反方向调整。

3.6 量子变异门

包含 Hadamard 门的量子遗传算法 (Hadamard quantum genetic algorithm, HQGA)^[16] 是通过量子变异门 (Hadamard 门) 对量子概率幅实现变异操作, Hadamard 门量子变异通过智能调整概率幅而非简单比特翻转,既能有效探索本地执行与边缘卸载的多种组合以避免陷入局部最优,又能维持种群多样性。其具体步骤包括:

- 1) 依据变异概率 M_r , 选中需要变异的量子比特。量子变异门的矩阵表示为:

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (21)$$

- 2) 只对选中的量子比特计算新的量子概率幅,公式为:

$$\begin{bmatrix} \alpha'_i \\ \beta'_i \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} \quad (22)$$

式中: $[\alpha_i \ \beta_i]^\top$ 为当前染色体的第 i 个基因; $[\alpha'_i \ \beta'_i]^\top$ 为当前染色体的第 i 个基因经过量子变异门更新后的新基因。

3.7 卸载策略设计

综上可知,QGA 可以解决单小区多用户下的总时延最小化问题,具体算法伪代码如算法 1 所示。

算法 1: 基于 QGA 算法的任务卸载策略	
输入: 任务数量 n 、种群大小 m 、最大迭代次数 $G_{\max}$ 、变异概率 M_r 、精英保留比例 E_r 、惩罚因子 λ	
输出: 最优卸载策略 S^{best}	
初始化量子种群,将种群中个体的全部量子比特进行初始化为 $1/\sqrt{2}$,并清空 S^{best}	
1) For $g = 1$ to $G_{\max}$ do	
量子测量:	
2) 对量子种群 $Q(t)$ 中所有量子比特根据式(15)测量得二进制种群 $P(t)$	
适应度评估:	
3) 对二进制种群 $P(t)$ 中每个个体根据式(17)评估适应度值,如果当前解优于历史最优解,则更新历史最优解	
精英保留:	
4) 根据式(18)选择二进制种群 $P(t)$ 中适应度最高的 E_{size} 个个体,替换下一代种群中最差的 E_{size} 个个体	
量子旋转门更新量子态:	
5) 对量子种群 $Q(t)$ 中所有量子比特根据式(21)进行量子旋转门更新,根据 S^{best} 和其适应度值动态调整旋转角,更新其对应的量子概率幅	
Hadamard 门变异:	
6) 以概率 M_r 随机对量子种群 $Q(t)$ 中量子比特应用 H 门进行变异	
7) 更新 S^{best}	
8) end for	

4 实验与分析

4.1 实验参数设置

本文采用 PyCharm 搭建仿真环境,本研究的仿真场景是在多用户单小区的网络模型下,该模型内含有 1 个配有 MEC 服务器的 BS。仿真实验涉及的参数参照文献[1]、[2]、[15]具体参数如表 2 所示。

4.2 仿真结果分析

本研究对比对象分 3 类:1)同场景传统方法(基于 GA 的卸载策略^[7]);2)同场景核心方法(基于 APSO 的卸载策略^[12]);3)算法改进验证组(TQGA、HQGA^[16])。

图 2 给出了不同任务卸载策略的算法下,不同任务数

表 2 仿真参数

Table 2 Simulation parameters

仿真参数	参数值
任务的数据量大小 B_i	7~40 Mbit
任务所需要的 CPU 周期 u_i	800~1 200 cycle/bit
移动设备的最大计算资源 $f_{\max}^{local}$	3 GHz
MEC 服务器的最大计算资源 $f_{\max}^{mec}$	20 GHz
信道带宽 W	1 MHz
移动设备与 MEC 服务器间的信道增益 H_i	$2 \times 10^{-10} \sim 2 \times 10^{-6}$
信道高斯噪声功率 N_0	1×10^{-9}
移动设备传输功率 P_i	0.5 W
移动设备空闲时传输功率 P_i^f	0.1 W
处理任务能接受的最大能耗 $E_i^{\max}$	5 J
惩罚因子 λ	0.6
种群大小 m	50
最大迭代次数 $G_{\max}$	100
变异概率 M_r	0.01
精英保留比例 E_r	0.1

据量的总时延。图中设置了惩罚因子为 $\lambda = 0.5$,计算任务数为 60,任务数据量(单位:Mbit)分别为 10、15、20、25、30、35、40。其中,总时延随着任务数量的增加而增加。在数据量 <20 时,提出的方法相较于其他方法有微弱的优势,但数据量不断增加时,所提出的方法优势越来越大,通过对比可以得出(以数据量 30 为例),QGA 在单小区多用户场景下的寻找最优卸载策略的总时延相比 GA 减少了 7.3%、相比 APSO 减少了 4.3%。

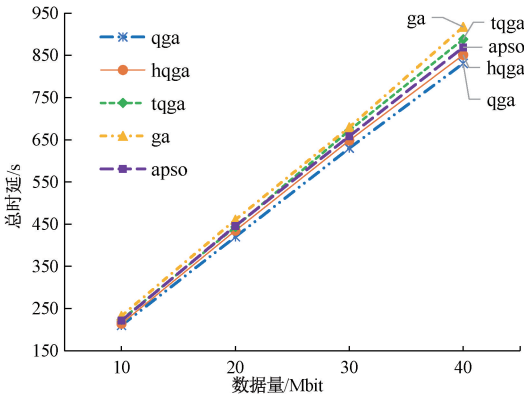


图 2 不同数据量的总时延

Fig. 2 Total delay for different data volumes

图 3 给出了在不同任务卸载策略算法下,不同计算任务数的总时延。图中设置惩罚因子为 $\lambda = 0.5$,任务数据量=20 Mbit,由于计算任务数较少时所有方法的区别不明显,所以对比计算任务数分别为 60、70、80、90、100 的结果。对 QGA、GA、HQGA、TQGA、APSO 的卸载策略的总时延

进行了比较。其中,计算时延随着任务数量的增加而增加,但任务越来越多时,QGA 的总时延优势愈发明显。

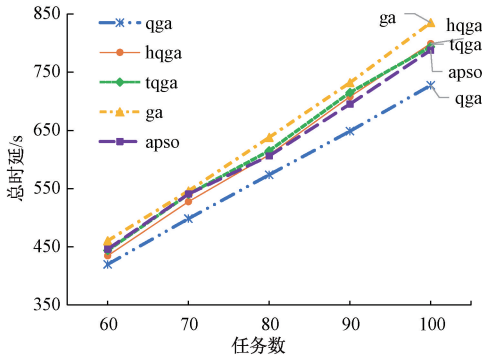


图 3 不同任务数的总时延

Fig. 3 Total delay for different numbers of tasks

图 4 给出了在不同任务卸载策略算法下,不同惩罚因子的总时延。图中设置计算任务数为 60,任务数据量为 20 Mbit 和能耗约束分别为 0.1、0.2、0.3、0.4、0.5、0.6、0.7、0.8。从图中可以得出,无论是从稳定性还是寻优能力上,所提方法均优于其他方法的。

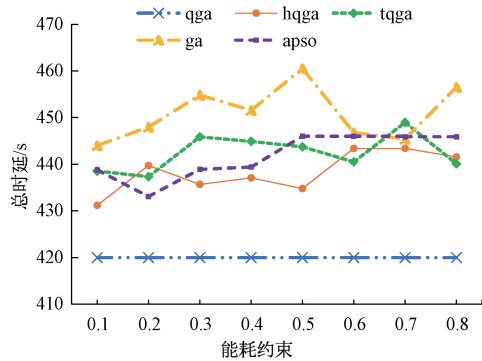


图 4 不同惩罚因子的总时延

Fig. 4 Total delay for different penalty factors

5 结 论

本研究聚焦 MEC 单小区多用户场景的任务卸载优化,以最小化计算任务总时延为核心目标,兼顾移动终端能耗、本地及 MEC 服务器计算资源约束,开展系统建模、算法设计与实验验证。首先,围绕总时延最小的卸载策略目标,搭建适配该场景的系统与计算模型;接着,融入多约束条件构建多约束下总时延最小化目标函数;最后,针对 TQGA 染色体多样性有限、收敛慢的问题,引入量子变异门与精英保留策略,形成改进算法以求解最优卸载策略。仿真实验将所提策略与其他策略在不同能耗约束、任务数度量及任务数场景下对比,结果显示其在各类场景中总时延均更小,验证了 QGA 的有效性与优越性。

参考文献

[1] ZHU AN Q, WEN Y Y. Computing offloading

strategy using improved genetic algorithm in mobile edge computing system[J]. Journal of Grid Computing, 2021,19:38.

[2] AI L H, TAN B, ZHANG J D, et al. Dynamic offloading strategy for delay-sensitive task in mobile-edge computing networks[J]. IEEE Internet of Things Journal,2023,10(1):526-538.

[3] HUO Q G, ZHANG W D, WU Z M, et al. Collaborative offloading strategy for dependent tasks in mobile edge computing [J]. Wireless Personal Communication, 2024, 134:267-292.

[4] 李亚,王卫岗,张原,等. 基于改进型 TD3 算法的车载边缘计算任务卸载决策[J]. 电子测量技术, 2024, 47(6):64-70.

LI Y, WANG W G, ZHANG Y, et al. Decision-making for vehicle-based edge computing task offloading based on the improved TD3 algorithm[J]. Electronic Measurement Technology, 2024, 47 (6): 64-70.

[5] 吴波,龙廷艳,万良,等. MEC 中基于改进粒子群算法的任务卸载策略[J/OL]. 计算机工程, 2025, 1-12 [2026-04-02]. <https://doi.org/10.19678/j.issn.1000-3428.0069852>.

WU B, LONG T Y, WAN L, et al. Task offloading strategy in MEC based on improved particle swarm algorithm[J/OL]. Computer Engineering, 2025, 1-12 [2026-04-02]. <https://doi.org/10.19678/j.issn.1000-3428.00698522>.

[6] LI W Z, TANG R, WANG X K, et al. A novel approach for computation offloading based on a parallel collaborative genetic algorithm in MEC[J]. Wireless Personal Communications, 2025,140:1119-1146.

[7] 王丽,王晓凯. 基于遗传算法的 MEC 任务卸载资源优化策略 [J]. 计算机工程与设计, 2023, 44 (10): 2909-2915.

WANG L, WANG X K. Optimization strategy for MEC task offloading resources based on genetic algorithm [J]. Computer Engineering and Design, 2023,44(10): 2909-2915.

[8] 丁嘉伟. 基于多变量联合优化的 MEC 网络能耗最小化算法[J]. 弹箭与制导学报,2023,43(3):39-44.

DING J W. A MEC network energy consumption minimization algorithm based on multivariable joint optimization[J]. Journal of Ballistics and Guidance, 2023, 43(3): 39-44.

[9] ZHUO ZH H, DONG SH,ZHENG H, et al. Method of minimizing energy consumption for RIS assisted UAV mobile edge computing system [J]. IEEE

- Access, 2024, 12: 39678-39688.
- [10] EANG C, ROS S, KANG S, et al. Offloading decision and resource allocation in mobile edge computing for cost and latency efficiencies in real-time IoT[J]. Electronics, 2024, 13(7): 1218.
- [11] 武晓雯, 郭孟营, 胡阿建, 等. 基于改进遗传粒子群算法的无人机路径规划[J]. 仪器仪表学报, 2025, 46(4): 315-325.
- WU X W, GUO M Y, HU A J, et al. Path planning for UAV based on improved hybrid genetic particle swarm algorithm[J]. Chinese Journal of Scientific Instrument, 2025, 46(4): 315-325.
- [12] 顾敏, 徐雅男, 王辛迪, 等. 多用户 MIMO-MEC 网络中基于 APSO 的任务卸载研究[J]. 无线电工程, 2024, 54(3): 711-718.
- GU M, XU Y N, WANG X D, et al. Research on APSO-based task offloading in multi-user MIMO-MEC networks[J]. Radio Engineering, 2024, 54(3): 711-718.
- [13] 辜航, 张敏娟, 潘钰文. 移动边缘计算中任务卸载和资源分配算法优化[J]. 现代电子技术, 2023, 46(7): 67-72.
- GU H, ZHANG M J, PAN Y W. Optimization of task offloading and resource allocation algorithms in mobile edge computing [J]. Modern Electronic Technology, 2023, 46(7): 67-72.
- [14] 赵婵婵, 杨星辰, 石宝, 等. 基于元强化学习的任务卸载优化策略[J]. 计算机科学, 2025, 52(S1): 836-843.
- ZHAO CH CH, YANG X CH, SHI B, et al. Task offloading optimization strategy based on meta reinforcement learning [J]. Computer Science, 2025, 52(S1): 836-843.
- [15] 占慧芳, 李正权, 武贵路, 等. D2D 辅助移动边缘计算下的任务卸载和资源分配研究[J]. 电子测量与仪器学报, 2025, 39(1): 90-100.
- ZHAN H F, LI ZH Q, WU G L, et al. Research on task offloading and resource allocation in D2D-assisted mobile edge computing[J]. Journal of Electronic Measurement and Instrumentation, 2025, 39(1): 90-100.
- [16] 王佳铭. 基于 Hadamard 门的量子遗传算法在 NP 问题中的研究及应用[D]. 青岛: 青岛理工大学, 2024.
- WANG J M. Research and application of quantum genetic algorithm based on Hadamard gate in NP problems [D]. Qingdao: Qingdao University of Technology, 2024.

作者简介

郭澳归(通信作者), 硕士研究生, 主要研究方向为移动边缘计算、任务卸载策略。

E-mail: 15771644715@163.com

叶成荫, 副教授, 主要研究方向为 Internet 的控制与决策、5G 网络优化、机器学习。

E-mail: chengyinye@foxmail.com

申佳欣, 硕士研究生, 主要研究方向为无线通信。

E-mail: shenjiaxin_2023@163.com