

# 控制与决策

Control and Decision

## 多无人机协同直播场景下自适应任务卸载决策

彭维平, 王明坤, 宋成, 贾宗璞

引用本文:

彭维平, 王明坤, 宋成, 等. 多无人机协同直播场景下自适应任务卸载决策[J]. *控制与决策*, 2021, 36(4): 974–982.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2019.1104>

---

## 您可能感兴趣的其他文章

Articles you may be interested in

### 面向多目标侦察任务的无人机航线规划

UAV trajectory planning for multi-target reconnaissance missions

控制与决策. 2021, 36(5): 1191–1198 <https://doi.org/10.13195/j.kzyjc.2019.1284>

### 输入受限的多旋翼无人机轨迹跟踪鲁棒正定不变集设计

Design of robust positively invariant set for trajectory tracking of multi-rotor UAV with input saturation

控制与决策. 2021, 36(4): 857–866 <https://doi.org/10.13195/j.kzyjc.2019.0880>

### 基于凸面体圆弧航路的无人机自主避障算法

Autonomous obstacle avoidance algorithm designed for UAV based on convex circular trajectory

控制与决策. 2021, 36(3): 653–660 <https://doi.org/10.13195/j.kzyjc.2019.0768>

### 四旋翼无人机抗干扰轨迹跟踪控制

Anti-interference trajectory tracking control of quadrotor UAV

控制与决策. 2021, 36(2): 379–386 <https://doi.org/10.13195/j.kzyjc.2019.0875>

### 微型无人机集群低时延组网规划方法

A low delay networking planning method for micro UAV swarm

控制与决策. 2020, 35(11): 2696–2706 <https://doi.org/10.13195/j.kzyjc.2018.1549>

# 多无人机协同直播场景下自适应任务卸载决策

彭维平<sup>†</sup>, 王明坤, 宋成, 贾宗璞

(河南理工大学 计算机科学与技术学院, 河南 焦作 454000)

**摘要:** 针对多无人机协同执行任务过程中计算量大和能耗高的问题, 基于计算卸载原理以及博弈理论, 提出一种多无人机自适应任务卸载方案. 在方案中首先对系统进行建模, 构造出多节点相互制约的移动卸载模型; 其次, 根据卸载模型分别构建无人机执行任务时的时延与能耗计算方法, 通过综合考虑延时和能耗两方面因素, 生成系统全局代价函数; 然后, 设计出基于博弈理论和纳什均衡的自适应任务卸载算法, 通过卸载算法与权重的分配实现最优计算节点的选取, 实现整个直播系统的代价最小, 从而平衡无人机计算时延与能量消耗; 最后, 与现有卸载模型相比, 所提出的方案在任务执行过程中具有较强的移动性, 能耗更低且时效性更高. 仿真结果验证了所提出理论的有效性, 具有现实意义.

**关键词:** 多无人机; 自适应; 博弈论; 任务卸载; 直播

中图分类号: TN929.5

文献标志码: A

DOI: 10.13195/j.kzyjc.2019.1104

开放科学(资源服务)标识码(OSID):



**引用格式:** 彭维平, 王明坤, 宋成, 等. 多无人机协同直播场景下自适应任务卸载决策[J]. 控制与决策, 2021, 36(4): 974-982.

## Adaptive task offloading decision of multi-UAVs cooperation in live broadcasting scenario

PENG Wei-ping<sup>†</sup>, WANG Ming-kun, SONG Cheng, JIA Zong-pu

(School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo 454000, China)

**Abstract:** In order to address the problem of large computation and high energy consumption in the process of multi-UAV collaborative missions, a multi-UAV adaptive task offloading scheme is proposed based on the computational offloading principle and game theory. The first step in the scheme is to model the system and construct a multi-node, mutually constrained mobile offloading model. Then, based on the unloading model, the method is constructed to calculate the time delay and energy consumption of the UAV when it performs its tasks respectively. The global cost function of the system is generated by considering both delay and energy consumption. An adaptive task offloading algorithm is designed based on the game theory and Nash equilibrium. The selection of the optimal computation node is achieved for minimizing the cost of the entire live system, thus balancing the UAV computation delay and energy consumption. Finally, compared with the existing models of offloading, task execution process of the proposed method has strong mobility, lower energy consumption and higher efficiency. The simulation results verify the effectiveness of the theory, which has a practical significance.

**Keywords:** multi-UAVs; self-adaptation; game theory; task; live

## 0 引言

随着无人机的应用领域越来越广泛, 应用环境也更加复杂多变, 单无人机技术得到了较快的发展, 但在续航能力、计算能力以及载重能力等方面均存在很大不足<sup>[1-2]</sup>, 采用多无人机协同是未来的发展趋势<sup>[3]</sup>. 国内外众多学者对其做了大量的研究工作, 主要集中在对无人机的协同控制<sup>[4-5]</sup>、航迹规划<sup>[6-7]</sup>和

任务分配<sup>[8-9]</sup>等方面, 对计算资源考虑较少. 在多无人机协同执行任务过程中随着无人机数量的增加, 计算量呈指数级增长, 但无人机自身的能量有限, 无法满足数据处理的实时性、低能耗的需求, 给多无人机协同发展带来严峻的挑战<sup>[10]</sup>. 将计算资源下沉到离无人机较近位置的边缘服务器上<sup>[11]</sup>, 每架无人机将收集到的信息以及计算任务发送给该服务器代理计

收稿日期: 2019-07-31; 修回日期: 2019-11-19.

基金项目: 河南省科技攻关项目(182102110333); 河南理工大学博士基金项目(B2012-050).

责任编辑: 张文安.

<sup>†</sup>通讯作者. E-mail: pwp9999@hpu.edu.cn.

算,能最大限度地降低计算时延,节省能耗,提高系统的运行效率。

近年来,许多学者开展了一系列的研究:Kim等<sup>[12]</sup>提出了一种自适应计算卸载的无人机监控系统,综合考虑了目标的移动性、计算能力、响应时间以及丢包率等各种因素,设计出一种动态计算卸载算法,降低了无人机对目标跟踪的反应时间;Cao等<sup>[13]</sup>提出了一种基于蜂窝网络无人机飞行系统,通过对无人机飞行轨迹的优化,在飞行过程中最大程度地接近地面基站,高效地卸载计算任务,提高了飞行性能;Messous等<sup>[14]</sup>提出了一种基于博弈思想优化的无人机边缘计算系统,利用本地计算或通过WIFI网络传输、移动网络传输策略,实现执行延时与能量消耗的平衡;Valention等<sup>[15]</sup>提出无人机群之间相互利用空闲资源来提高计算能力,从而提高飞行时间,减小响应时间;Wang等<sup>[16]</sup>提出了一种基于边缘计算的无人机直播方案,在无人机通信速率较差而不足以支持视频播放的情况下,通过中继无人机增强无线网络的通信速率,提高视频质量;Chen等<sup>[17-18]</sup>针对多移动用户的计算卸载问题,基于博弈模型的结构性质,提出一种分布式计算卸载算法以提高计算卸载的效率;Deng等<sup>[19]</sup>基于博弈论的思想,针对多用户任务迁移问题,提出了一种分布式任务迁移决策方案以减小任务卸载的开销。考虑到以上方案在自适应选择卸载对象时存在的不足,本文在前人的研究基础之上,基于视觉定位与跟踪技术<sup>[20-21]</sup>,提出一种多无人机直播体育赛事的任务卸载方案,设计一种应用于小规模集群的轻量级算法,以解决无人机在直播过程中任务量大的问题,最大限度地减少能量消耗,降低响应时间,从而提高集群无人机群的生存时间以及执行任务的能力。

## 1 系统模型

### 1.1 场景分析

多无人机直播足球比赛的应用场景如图1所示。本场景下,所有无人机相互实时共享位置及状态信息,相互不发生碰撞。主无人机 $j_5$ 通过计算机视觉技术实现对足球的识别与跟踪,计算任务可选择较优的计算资源(边缘服务器),其附属无人机 $\{j_1, j_2, j_3, j_4, j_6\}$ 采集现场的其他主要视频信息。当足球越出所有无人机的视觉范围时,整个系统进入对足球搜索阶段,即所有无人机分析各自的视频信息对足球作出大致的定位,经过综合分析计算出足球的精确位置,然后指派最近的无人机作为主机实现对足球的跟踪。

对上述场景抽象出数学模型如下:定义无人机

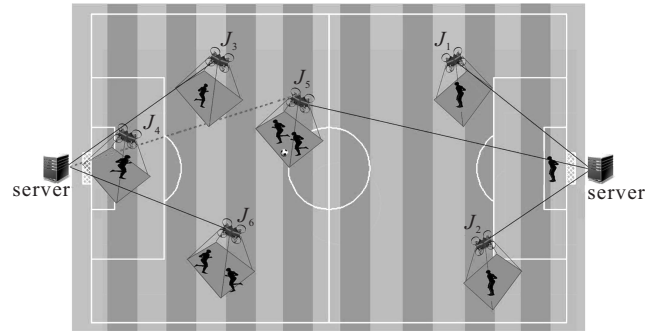


图1 直播场景

集合 $N_{\text{drone}} = \{j_1, j_2, \dots, j_N\}$ ,  $S$ 为边缘服务器节点。所有节点的计算能力均采用完成单位任务量所需的CPU周期来表示。假设边缘服务器的计算能力为 $F_s$ ,移动节点的计算能力为 $f_j$ 。某一视频信息分析所需的计算资源定义为三元组 $W = \{R, D, T_{\text{max}}\}$ ,其中: $R$ 表示计算任务, $D$ 表示传输的数据大小, $T_{\text{max}}$ 表示为达到视频直播要求而处理该任务量所能承受的最大延时。在上述场景中,多个相近的移动节点可进行自组网,其计算任务可在本地执行也可选择卸载到 $S_1$ 或 $S_2$ 服务器执行。定义 $M_0, M_1, M_2$ 分别为选择本地计算、卸载到 $S_1$ 、卸载到 $S_2$ 所有节点的集合。对应的元素个数分别为 $m_0, m_1, m_2$ 且满足 $m_0 + m_1 + m_2 = N$ 。由于运算结果数据量较小且下行数据量较大,本文不考虑回传结果的延时,为减少由于算法迭代带来额外的开销,采取间断的方式进行卸载判断。

### 1.2 通信模型

无人机直播场景必须有足够的网络资源以实现视频数据及控制数据的实时传输。设边缘计算服务器 $S$ 的水平位置坐标 $G_n$ 不变,分别表示为 $G_1(x_1, y_1), G_2(x_2, y_2)$ 。无人机 $j_n$ 离地面的高度为 $h$ ,水平位置坐标表示为 $J_n(x_0, y_0)$ ,则 $j_n$ 到服务器 $M_1, M_2$ 的距离可以表示为

$$d_{j_n, G_n} = \sqrt{h^2 + \|J_n - G_n\|^2}, \quad (1)$$

其中 $\|*\|$ 为欧几里得范数,表示两点间的距离。节点移动的最大速度为 $V$  m/s,由文献[20],无线网络中节点到服务器的增益 $H$ 可以表示为

$$H = \frac{\beta}{d_{j_n, s}^2} = \frac{\beta}{h^2 + \|J_n - G_n\|^2}, \quad (2)$$

其中 $\beta$ 为单位距离的网络增益。由文献[22-23],节点在某一位置与边缘计算服务器 $M$ 的传输速率可以表示为

$$C_{j_n, s} = B \log_2 \left( 1 + \frac{PH}{\sigma^2} \right) = B \log_2 \left( 1 + \frac{\rho}{h^2 + \|J_n - G_n\|^2} \right). \quad (3)$$

其中:  $\rho = P\beta/\sigma^2$  为信噪比,  $d$  为节点与服务器之间的距离,  $B$  为网络带宽,  $P$  为节点的发射功率,  $\sigma^2$  为噪声功率. 在可分的一时间段内传输的数据速率为  $C_{j_n,s}$ , 则传输时间可以表示为

$$T_{j_n}^{\text{tran}} = \frac{uD}{C_{j_n,s}}. \quad (4)$$

其中:  $u$  为数据参量,  $D$  为传输数据量的大小. 所要消耗的能量为

$$E_{j_n}^{\text{tran}} = P_{j_n} T_{j_n}. \quad (5)$$

### 1.3 计算模型

设节点  $j_n$  的计算能力为  $f_l$ , 边缘计算中心总的计算能力为  $F_s$ , 且均保持不变.

#### 1.3.1 卸载到计算中心

当节点  $j_n$  将任务卸载到服务器上时, 所需的计算时间可以表示为

$$T_s = \frac{vR}{F_s}, \quad (6)$$

其中  $v$  为计算参量. 卸载到服务器所需要的时间为传输时间与服务器的处理时间之和, 即

$$T_{j_n}^{\text{sum}} = T_{j_n}^{\text{tran}} + T_s. \quad (7)$$

由于边缘计算中心有充足的能量, 不考虑其能量的消耗, 卸载任务所需要的总能量消耗为传输数据所消耗的能量, 可表示为

$$E_{j_n}^{\text{tran}} = P_{j_n} T_{j_n}^{\text{tran}}. \quad (8)$$

#### 1.3.2 本地计算

本地运行所消耗的时间为节点的计算时间, 即

$$T_{j_n}^{\text{sum}} = \frac{R}{f_l}. \quad (9)$$

当任务量在本地执行时, 所消耗的能量为传输数据的能耗与本地计算所消耗的能量之和. CPU 处理任务  $R$  所消耗的能量可表示为

$$E_{j_n}^l = R\delta. \quad (10)$$

其中:  $\delta$  为一个 CPU 计算周期所消耗的能量, 且  $\delta = \lambda f_l^2$ ;  $\lambda$  的大小取决于 CPU 的物理结构, 为固定值. 根据文献 [24], 设定  $\lambda = 10^{-27}$ , 本地计算所消耗的总能量可表示为

$$E_{j_n}^{\text{sum}} = E_{j_n}^{\text{tran}} + E_{j_n}^l. \quad (11)$$

### 1.4 效用函数

综合考虑时延和能耗两方面的因素, 任务执行所需的代价通过以下效用函数进行衡量:

$$\text{Cost}_{j_n} = \gamma_{j_n} T_{j_n}^{\text{sum}} + \eta_{j_n} E_{j_n}^{\text{sum}}. \quad (12)$$

其中:  $\text{Cost}_{j_n}$  为执行计算任务的总代价参量;  $\gamma_{j_n}$  和  $\eta_{j_n}$  分别表示节点  $j_n$  时延和能耗的权重, 满足  $\gamma_{j_n} +$

$\eta_{j_n} = 1$ . 权重代表系统对时延或能耗的偏好, 根据对直播的要求可以设定不同的权重值, 在时延与能耗之间进行权衡. 当要求低延时, 要增大延时权重; 当要求节省更多的能量时, 就要增加能耗的权重. 在实际应用中, 无人机装载电源的性能会有差异, 当电池能量较充足时, 可以适当增加时延的权重, 提高无人机跟踪目标的灵敏度, 提升用户体验; 当电池容量受限时, 可以增加能耗权重, 以满足实时直播任务的情况下实现能量消耗最小.

### 1.5 卸载模型

移动节点在移动过程中, 其通信资源和计算资源会时刻发生变化, 数据处理任务选择在本地执行还是在边缘计算服务器执行是卸载模型要解决的关键问题. 如果有太多的移动节点同时通过同一网络传输数据到边缘服务器, 则会带来较大的传输延时, 消耗更多的能量. 由于节点在运动过程中离边缘计算中心的距离在不断变化, 卸载集合也会随着场景的改变而动态变化, 选择代价较小的卸载对象是最优的卸载决策.

当处理某一任务量  $W$  所需的总延时大于节点处理该任务限定的最大允许延时  $T$  时, 为满足视频直播的需求; 首先需要改变卸载对象, 从而满足延时的需求; 而当总延时小于最大延时  $T$  时, 就需要综合考虑能耗, 利用效用函数进行卸载决策的选择, 以实现卸载代价最小.

本文设计以下卸载模型. 首先, 将整个直播时长平均分为多个时段  $T_{\text{slot}}$ , 经过一个时段, 卸载集合中的节点会根据对花费代价的预估赋予卸载决策的值, 从而加入相应的卸载集合, 并从原卸载集合中退出, 如图2所示.

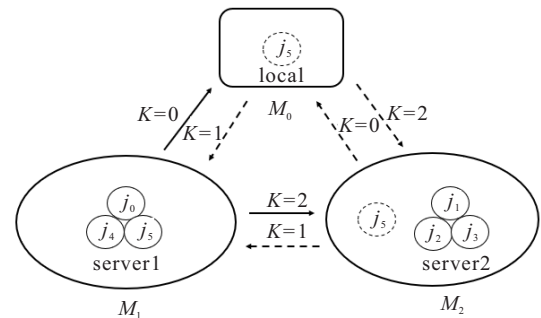


图2 移动卸载模型

图2中  $K$  为决策空间, 且

$$K = \begin{cases} 0, & \text{本地执行;} \\ 1, & \text{卸载到 } S_1; \\ 2, & \text{卸载到 } S_2. \end{cases}$$

此模型中, 同一个卸载集中的节点共享通信资源

和计算资源,如果大量的移动节点选择同样的卸载对象  $M$ ,则会给服务中心带来较大的计算压力,降低数据的传输速率. 如果传输速率较低,则会产生较大的能量消耗,甚至不能满足视频的传输需求. 同样,计算资源不足时,计算任务需排队等待,就会产生较大的计算延时. 为解决这一问题,本文用博弈的思想解决相互关系的问题,确保在某一状态下每一节点的决策都为最优.

## 2 基于博弈论的计算任务卸载

### 2.1 博弈的思想

博弈论是在存在多方主体的情况下,如何选择一种策略使得每个主体方获利均为最大的一种数学方法. 在博弈中主体之间的博弈是相互的,每一主体的决策都会对其他局中主体产生影响. 博弈的目的是在各个局中人充分理性的情况下,权衡每个局中人的利益,实现每个局中人的利益最大化.

### 2.2 博弈的数学模型

博弈的数学模型主要包括局中人、策略集和效用函数3个元素,表示为  $\xi(N, I, \partial)$ ,  $N$  为局中人的集合,  $I$  表示局中人卸载纯策略的集合,  $\partial$  为这一局势下的策略组合效用函数.

1) 局中人:也称决策者,在博弈时可以决定自己的行动方案的个人或团体. 博弈的一个特点就是每个局中人都是“理智的”,即每个局中人都会根据其他局中人的决策选择自己最合适的策略,以此来最大化自身利益且不存在侥幸心理(认为其他局中人做出失误的决策).

2) 策略集:博弈中供局中人选择决策的集合,包含局中人在某一局势下所有的决策. 策略集分为纯策略集和混合策略集. 在给定信息下,若只能在策略集中选择特定的决策,则称此策略集为纯策略集;若以某种概率在策略集中选择,则称此策略集为混合策略集.

3) 效用函数:表示在给定的局势下某一可以比较的值与局中人收益的映射关系,效用函数是博弈局势中作出决策的根本依据.

在整个博弈过程中,局中人  $r$  选择策略集合为  $I_r$ , 其中  $I_r$  中的一个决策为  $a_r$ , 则局中人  $r$  的决策集合可以表示为  $I_r(a_1, a_2, \dots, a_{r-1}, a_r, a_{r+1}, \dots, a_m)$  或  $(a_r, a_{-r})$ , 其中  $a_{-r} = (a_1, a_2, \dots, a_{r-1}, a_{r+1}, \dots, a_m)$  表示除节点  $r$  的决策  $a_r$  之外的其他局中人决策的集合. 在整个任务执行周期中,每一个局中人所作的决策是随着局势的变化而改变的,所付出的代价用效用函数表示,则局中人  $r$  的效用函数为  $\partial_r(a_r, a_{-r})$  或者

$$\partial_r(a_1, a_2, \dots, a_{r-1}, a_r, a_{r+1}, \dots, a_n).$$

### 2.3 最优决策建模

无人机集群在某一局势中的决策空间为  $K$ , 移动节点  $j_n$  在某一状态下给定一个策略集  $k_{-j_n}$ , 于是  $j_n$  就能根据  $k_{-j_n}$  赋予决策变量  $k_{j_n}$  最优的值,即选择最优决策来实现卸载开销最小. 代价函数可表示为

$$\partial_{j_n}(k_{j_n}, k_{-j_n}) = \begin{cases} \gamma_{j_n} T_{j_n}^l + \eta_{j_n} E_{j_n}^l, & k_{j_n} = 0; \\ \gamma_{j_n} (T_{j_n, s_1}^{\text{tran}} + T_{j_n, s_1}) + \eta_{j_n} E_{j_n, s_1}^{\text{tran}}, & k_{j_n} = 1; \\ \gamma_{j_n} (T_{j_n, s_2}^{\text{tran}} + T_{j_n, s_2}) + \eta_{j_n} E_{j_n, s_2}^{\text{tran}}, & k_{j_n} = 2. \end{cases} \quad (13)$$

对于节点的卸载代价可表示为

$$\partial_{j_n}(k_{j_n}) = \min \partial_{j_n}(k_{j_n}, k_{-j_n}), \quad \forall j_n \in N, k_{j_n} \in \{0, 1, 2\}, T_{j_n}^{\text{sum}} \leq T_{j_n}^{\text{max}}. \quad (14)$$

$T_{j_n}^{\text{sum}} \leq T_{j_n}^{\text{max}}$  说明单位任务量的处理时间不能大于最大延时  $T_{j_n}^{\text{max}}$ ,  $k_{j_n}$  的取值即为  $j_n$  在这一局势下的最优决策.

### 2.4 基于博弈论的任务卸载原理

博弈的结果是最终要达到纳什均衡,其定义如下.

**定义1(纳什均衡)** 在  $N$  个局中人的博弈  $\xi = \{I_1, I_2, \dots, I_N; \partial_1, \partial_2, \dots, \partial_N\}$  中,对于任意局中人  $r$  在其他局中人决策  $i_{-r}^* = (i_1^*, i_2^*, \dots, i_{r-1}^*, i_{r+1}^*, \dots, i_N^*)$  都为上策的情况下,  $i_r^*$  是  $r$  作出的最优决策. 若满足

$$\partial_r(i_r^*, i_{-r}^*) \leq \partial_r(i_r, i_{-r}^*), \quad \forall i_r \in I_r, \forall r \in N, \quad (15)$$

则称  $\partial_r(i_r^*, i_{-r}^*)$  为博弈的一个纳什均衡.

本文直播过程中,将某一个状态(即所有无人机在每一时隙的位置及卸载决策)作为一个局势,局中人集合  $N$  的元素为  $n$  架无人机,每个局中人在某一状态所作的策略集合为  $I$ , 这一状态下采用的决策代价的集合为  $\partial$ , 由局中人、策略集和代价函数可得到博弈的基本模型  $\xi(N, I, \partial)$ . 卸载模型时隙图如图3所示,在某一时隙  $\Delta t$  中存在时刻  $t$ , 满足  $t \in \Delta t$ , 当时隙  $\Delta t$  足够小时,集群无人机达到一个静止的状态. 在时刻  $t$ , 无人机  $j_n$  预估卸载代价来判断当前决策是否为最优,若不是最优决策,则执行卸载更新. 在时隙  $\Delta t$

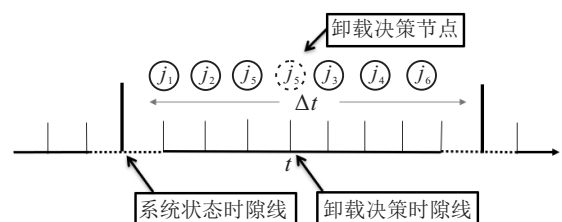


图3 卸载模型时隙图

中,所有无人机循环执行这一过程,直到每架无人机的决策都为最优。

由文献[18]可知,在博弈过程中,对于 $\forall i_r \in I_r, \forall r \in N, I$ 为策略集,如果效用函数 $\partial_r(i_r^*, i_{-r}^*) \leq \partial_r(i_r, i_{-r}^*)$ ,满足方程 $\phi(i_r^*, i_{-r}^*) \leq \phi(i_r, i_{-r}^*)$ ,则称此博弈为势博弈。并且,势博弈最少存在一个纳什均衡<sup>[25]</sup>。很显然,本文视频直播场景满足

$$\begin{aligned} & \phi(k_j^*, k_{-j}^*) - \phi(k_j, k_{-j}^*) \partial = \\ & \partial_j(k_j^*, k_{-j}^*) - \partial_j(k_j, k_{-j}^*), k \in \{0, 1, 2\}, j \in N. \end{aligned} \quad (16)$$

其中: $\phi(k_j^*, k_{-j}^*) = \min \partial_j(k_j, k_{-j}) = \varphi(k_j^*), \varphi(k_j^*)$ 是节点 $j$ 作出最优决策 $k_j^*$ 时的代价最小值。因此,对于视频直播博弈模型代价函数 $\partial_j(k_j, k_{-j})$ 而言,满足势函数方程 $\phi(k_j^*, k_{-j}^*) \leq \phi(k_j, k_{-j}^*)$ ,是势博弈。

由文献[18, 25]可知,本场景在有限博弈(有限的策略空间)下存在纳什均衡且唯一,即每架无人机的决策都为上策,达到纳什均衡。

节点 $j_n$ 为移动节点,即 $j_n \in N$ ,整个任务执行过程中的决策空间为 $K$ ,每隔时间 $T_{\text{slot}}$ ,无人机在某一状态下执行卸载更新算法,节点 $j_n$ 对执行任务代价进行预估。若决策 $k_{j_n}$ 满足

$$\partial_{j_n}^t(k_{j_n}) = \min \partial_{j_n}^t(k_{j_n}, k_{-j_n}), t \in \Delta t, \quad (17)$$

则移动节点 $j_n$ 的决策不变;若不满足,则执行一次卸载更新,并记录 $k_{j_n}$ 的值。

### 3 卸载决策算法设计

#### 3.1 系统初始化

根据实际应用场景,首先设定无人机运动区域 $Z$ 的覆盖范围,主要包括长度、宽度以及无人机飞行高度。其次,根据区域 $Z$ 的大小,设定每个无人机节点的初始位置的三维坐标,并通过坐标分别划分卸载集合 $M_0$ 、 $M_1$ 和 $M_2$ 。每个节点按照各自所在的卸载集,通过WIFI连接到相应的服务器,形成集中式网络。根据卸载集内节点的数量,分别对移动总节点数 $n$ 以及参数 $m_0$ 、 $m_1$ 和 $m_2$ 赋值,设定卸载决策 $k \in K = \{0, 1, 2\}$ 。节点 $j_n$ 的卸载决策 $k$ 的值0、1、2分别对应的集合为 $M_0$ 、 $M_1$ 和 $M_2$ 。节点之间通过D2D进行分布式连接,组成网状网络。根据节点及服务器性能,设定节点 $j$ 与服务器 $S$ 的计算能力为 $f_j$ 和 $F_s$ ,初始化权重 $\eta$ ,且 $\gamma = 1 - \eta$ ;节点移动最长时间 $T_{\text{move}}^{\text{max}}$ (min)以及卸载判断间隔时间 $T_{\text{slot}}$ (min);处理任务量 $R$ 所需时间 $T$ 的最大值 $T_{\text{max}}$ 为单位时间;设置初始时间 $T_{\text{move}} = 0$ 。此时整个系统在时隙 $\Delta t$ 中处于静止状态,即达到一个局势 $\xi$ ,其中 $\lim \Delta t = 0$ 。整个系统调整到最优状

态在一个时隙 $\Delta t$ 内完成。

#### 3.2 任务卸载决策

根据2.4节任务卸载原理,在一个确定的局势 $\xi_0$ 下,通过优化后的任务卸载算法可使每个节点 $j$ 的决策 $k$ 都为最优,具体步骤如下。

step 1: 在初始状态下,节点 $j$ 根据自己的计算能力计算出单位任务量的计算延时 $T_j^{\text{sum}}$ 及能耗 $E_j^l$ 。

step 2: 所有节点 $j$ 执行卸载算法,根据 $\eta$ 的取值,由代价函数得出在某一时刻 $\Delta t$ 内本地计算最小代价 $\partial_{k=0}^j(\eta, \Delta t)$ 。

step 3: 在 $k = 1$ 或 $2$ 时,节点根据位置坐标及式(1)计算到服务器的距离,由距离以及卸载集 $M$ 中的元素数 $m$ ,根据式(3)和(4)预估通信速率 $C$ 。

step 4: 根据集合 $M$ 中的节点数 $m$ 以及 $k$ 的取值,由式(6)计算单位任务的计算延时 $T_s$ ,由通信速率计算单位任务量的通信延时 $T_{\text{tran}}$ 以及通信能耗 $E_{\text{tran}}$ 。当 $k = k_n$ 时, $k_n \in \{1, 2\}$ 。由式(14)的约束条件,若 $T_s + T_{\text{tran}} < T_{\text{max}}$ ,则设置 $k \neq k_n$ 。

step 5: 根据2.4节任务卸载原理、卸载决策 $k$ 的值以及 $\eta$ 的取值,由式(13)计算出任务卸载代价,得出决策 $k$ 下的 $\min \partial_k^j(\eta, \Delta t)$ ,比较 $\min \partial_{k=0}^j(\eta, \Delta t)$ 与 $\min \partial_k^j(\eta, \Delta t)$ 的大小,得出在最优决策下节点 $j$ 的最小代价 $\min \partial_{k_n}^j(\eta, \Delta t)$ 。

step 6: 若节点 $j$ 的决策变量初始值 $k$ 满足 $\partial_k^j(\eta, \Delta t) = \min \partial_{k_n}^j(\eta, \Delta t)$ ,即初始决策为最优决策,则 $k$ 的值不更新;若满足 $\partial_k^j(\eta, \Delta t) > \min \partial_{k_n}^j(\eta, \Delta t)$ ,则设置 $k = k_n$ ,同时更新卸载集合。

step 7: 对于 $\forall j$ 在时隙 $\Delta t$ ,每个节点决策变量初始值 $k$ 都满足 $\partial_k^j(\eta, \Delta t) = \min \partial_{k_n}^j(\eta, \Delta t)$ ,则系统达到纳什均衡,遍历结束,执行step 8;否则,若 $\exists j$ 满足 $\partial_k^j(\eta, \Delta t) > \min \partial_{k_n}^j(\eta, \Delta t)$ ,此时节点需加入新的集合,同时更新决策 $k$ 的值及卸载集合 $M$ ,则所有节点需重新执行卸载算法,转到step 2。

step 8: 各节点执行随机运动算法,经过时间间隔 $T_{\text{slot}}$ ,在时隙 $\Delta t$ 中,各节点在这一态势下形成另一个局势 $\xi_n$ ,并更新运动总时间 $T_{\text{move}}$ 的值。若 $T_{\text{move}} \leq T_{\text{move}}^{\text{max}}$ ,则执行step 2;若 $T_{\text{move}} > T_{\text{move}}^{\text{max}}$ ,则执行step 7。

step 9: 程序执行完毕,退出系统。

### 4 仿真实验与分析

#### 4.1 实验环境

根据实际应用场景,本次实验设定无人机 $j$ 运动区域 $Z$ 的覆盖范围为:长度110m,宽度66m,高6m。由于自适应卸载算法会根据局势自动选择卸载

对象,初始状态下节点可以放在任意卸载集合中.为确保无人机节点在足球场均匀分布,本次仿真实验选择卸载集内的节点及初始位置的三维坐标为

$$M_1 : \{G_1(-60, 33, 0), j_0(-15, 33, 6), j_4(-45, 44, 6), j_5(-45, 22, 6)\},$$

$$M_2 : \{G_2(60, 33, 0), j_1(15, 33, 6), j_2(45, 44, 6), j_3(45, 22, 6)\};$$

$M_0 = \{ \}$ . 其中:集合  $M_0$  为空集,无人机  $J$  的高  $h = 6\text{ m}$  为定值.仿真实验环境为:使用 Intel i5-8 250u 处理器, Window10 操作系统,采用 Java SE 开发工具.实验中使用 WIFI 通信技术,遵循 IEEE 802.11 通信协议, 2.4 GHz 频段,设定无线通信的带宽为 50 MHz,每个节点  $j$  单位时间内需传输的任务量  $D = 3.8\text{ Mbps}$ . 节点的计算能力与服务器的计算能力分别为  $f_{\text{local}} = 2\text{ GHz}$  和  $F_s = 20\text{ GHz}$ ,以满足视频处理的需求.单位时间内每一节点的任务量为  $R = 1.8 \times 10^9$  个 CPU 周期,6 个节点初始位置均匀分布在高为 6 m、长为 110 m、宽为 66 m 的平面区域中,所有节点以 15 m/s 的速度随机运动 45 min,即  $T_{\text{move}}^{\text{max}} = 45(\text{min})$ ,卸载判断时间间隔  $T_{\text{slot}} = 5(\text{min})$ ,权重取值集合为  $\eta \in \{0, 0.2, 0.4, 0.5, 0.6, 0.8, 1\}$ .

#### 4.2 实验结果与分析

为验证所提出的基于博弈理论的任务卸载算法的优越性,结合实际场景分别考虑以下3种情况:

- 1) 本地计算:所有节点计算任务全部都在本地执行;
- 2) 任务卸载:所有节点连接到一个服务器,整个过程由特定的边缘服务器执行计算任务;
- 3) 博弈选择:所有节点按照本文提出的场景与算法选择执行决策,自适应地选择计算资源丰富的服务器.

本次仿真从延时、能耗和代价3个参量进行分析.每一节点在整个过程中,在规定的时间内进行卸载判断,并避免频繁进行卸载判断所带来自身额外的代价开销.

##### 4.2.1 延时分析

在节点随机运动 45 min 的运动轨迹中,每隔 5 min 取出位置坐标,然后分析单位任务量计算延时的变化,如图4所示.

本文提取出同一时刻下初始位置较为典型的3个节点的计算延时.从3个节点的时间数据上看,在本地执行计算任务时都为定值 0.9 s,不受本区域内其

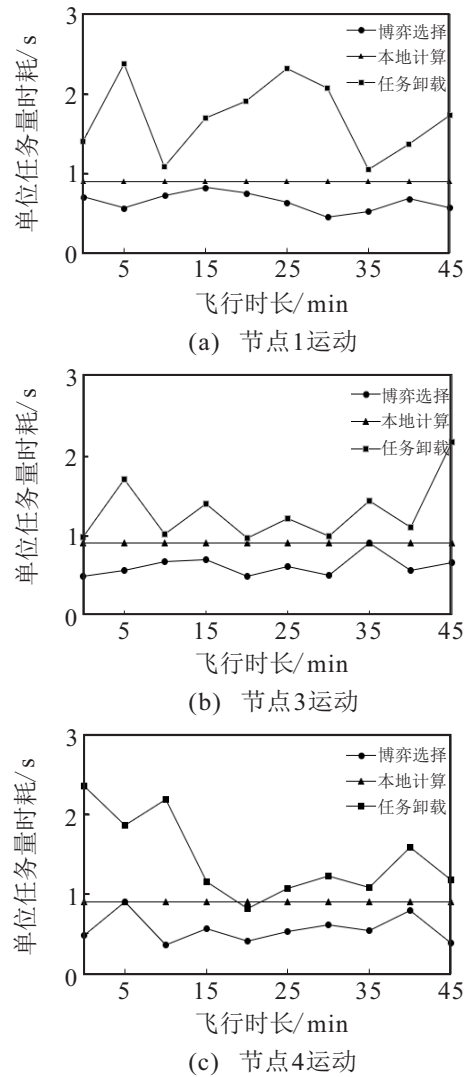


图4 单位任务量计算延时

他节点的影响和节点位置改变的影响,这是因为本地计算过程中节点之间没有共享计算资源和通信资源,计算延时一直处于稳定状态.任务卸载到一个服务器的情况从数据上看几乎全都大于本地计算延时,而且在某些时刻远远大于本地计算延时,这是因为所有节点共享同一通信资源和计算资源,网络负载与计算中心负载都较大,所以引起计算延时较大.在计算中心计算任务时,计算延时应趋于稳定状态,而图4数据显示延时变化较大,因此,节点位置对计算延时的影响处于主导地位.节点4在第20 min这一时刻卸载计算延时低于本地计算延时,这说明节点处于离计算中心较近的位置,在整个群集节点中这样的时刻出现的概率较小.博弈选择卸载策略是3种计算方式延时最低的一种,在整个过程中所有节点都选择自己最优的决策,即整个系统达到一个最优的局势,满足视频处理低延时的要求.节点3在第35 min,节点4在第5 min这一时刻的计算延时都等于本地计算延时,说明这一时刻计算任务选择在本地计算.各节点在整

个运动过程中的平均时延如图5所示.

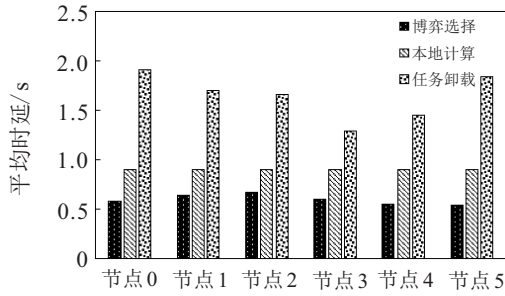


图5 各节点单位任务量时延

在本地计算是较为稳定的,但整体时延高于博弈选择时延. 博弈选择情况下相对稳定且延时较低,能满足实时性的要求. 情况最不好是任务卸载,受节点移动影响较大且延时较大,节点5最高延时比博弈选择延时高70%.

### 4.2.2 能耗分析

以下分析从节点能耗的角度进行,与考虑延时的同一节点的同一轨迹节点能耗进行对比分析,如图6所示.

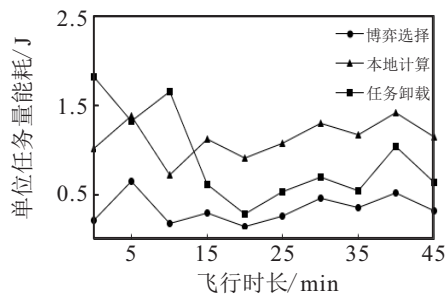
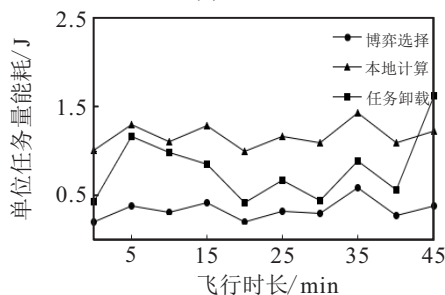
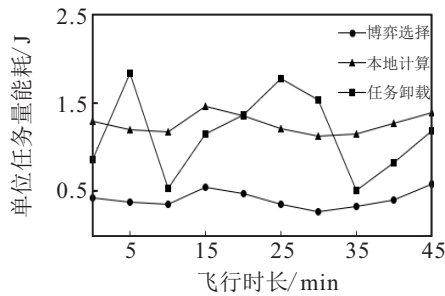


图6 单位任务量消耗的总能量

从节点能耗数据来看,本地计算能耗较大,且在很多情况下大于任务卸载的能耗,这是因为在节点计

算任务的同时,还需要发送视频数据到服务器以实现直播功能,但由于低延时的需求,计算任务必须在本地处理以保证系统的正常运行,相比于任务卸载能耗,本地计算能耗波动较小;本地计算能耗相比于本地计算延时,能耗有波动且相似于博弈选择能耗,主要是由传输视频数据时能耗变动引起的. 任务卸载的能耗,受节点到服务器距离的影响较大,例如节点1在第5 min这一时刻,单位任务量的能耗达到1.84 J,说明节点离服务器的距离最远. 各节点在整个运动过程中的平均能耗如图7所示.

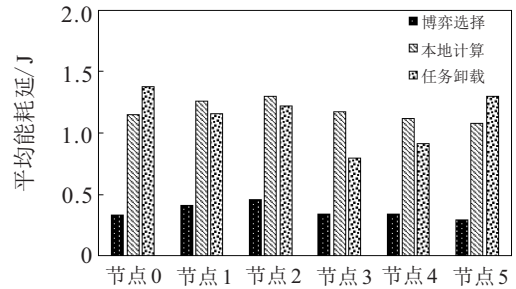


图7 各节点平均能耗分析

很多情况下节点选择在本地计算的能耗会大于任务卸载,且相对稳定. 博弈选择策略在能耗方面占较大优势,节点5的任务卸载能耗最高比博弈选择高77%.

### 4.2.3 代价分析

所有节点在整个运动过程中,不同权重对应的平均计算代价如表1所示.

表1 各节点不同权重下的代价值

对象	指标	代价权重 $\eta$				
		0	0.2	0.5	0.8	1
节点0	博弈选择	0.56	0.53	0.49	0.44	0.36
	本地计算	0.9	1.0	0.99	1.09	1.15
	任务卸载	1.84	1.75	1.71	1.49	1.38
节点1	博弈选择	0.64	0.65	0.59	0.46	0.41
	本地计算	0.9	0.96	1.03	1.18	1.26
	任务卸载	1.72	1.59	1.48	1.26	1.16
节点2	博弈选择	0.67	0.66	0.60	0.48	0.44
	本地计算	0.9	0.97	1.04	1.19	1.17
	任务卸载	1.67	1.56	1.45	1.24	1.13
节点3	博弈选择	0.60	0.59	0.53	0.42	0.34
	本地计算	0.9	0.95	1.01	1.13	1.17
	任务卸载	1.29	1.18	1.07	0.86	0.75
节点4	博弈选择	0.55	0.57	0.52	0.41	0.35
	本地计算	0.9	0.94	0.99	1.09	1.14
	任务卸载	1.45	1.34	1.23	1.01	0.91
节点5	博弈选择	0.54	0.52	0.47	0.46	0.29
	本地计算	0.9	0.94	0.99	1.09	1.15
	任务卸载	1.84	1.74	1.63	1.41	1.30

为便于分析,取出3个典型的权重值进行对比分析,如图8所示.

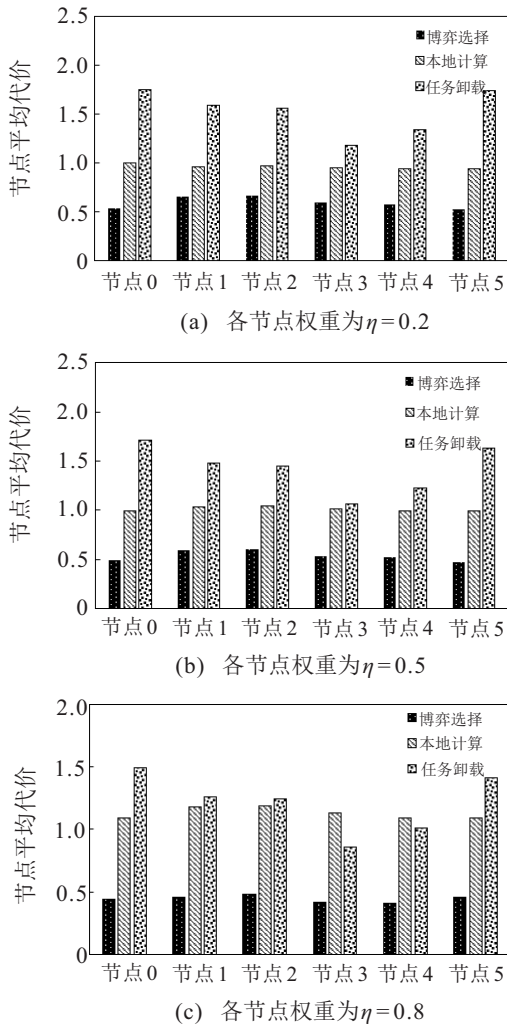


图8 不同权重下各节点单位任务量平均代价

由图8中的数据分析可知:本地计算随着权重 $\eta$ 值的增大而变大且越来越不稳定;任务卸载虽然代价有所降低,但稳定性得不到保障.根据效用函数分析,权重 $\eta$ 的值越大,节点的代价就会越小,系统主要偏好是低能耗,但延时相对变大;相反,权重 $\eta$ 的值越小,系统偏好为低延时,能耗相对变大.对不同节点之间分析,博弈选择在平衡时延与能耗时更具优越性及稳定性,受节点位置移动影响较小,实现负载均衡,避免节点的单点压力过大而耗尽资源.

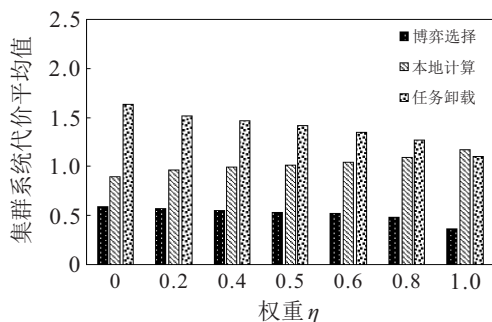


图9 所有节点不同权重平均代价值

以下给出整个过程中所有节点基于博弈算法自适应选择卸载对象过程中权重 $\eta$ 取不同的值所对应的平均系统代价,如图9所示.

### 5 结论

本文针对多无人机直播足球比赛的应用场景,利用边缘计算理论,提出了一个基于博弈理论和纳什均衡的自适应任务卸载方案,并通过仿真实验模拟了一场6架无人机直播上半场足球比赛.实验结果表明,本地计算能耗较大,任务卸载计算延时较高且不稳定.相比之下,基于博弈理论的任务卸载方法优势较为明显,无论是在计算延时方面还是能量消耗方面,都具有较高的优越性,并能适应不同权重的选择,实现了负载均衡,避免单点压力过大,最大程度地减小集群无人机之间的相互影响和由于无人机的移动而产生的影响,具有较强的稳定性及适应性,为集群式环境下的任务卸载提供了一定的参考.

### 参考文献(References)

- [1] 宗群,王丹丹,邵士凯,等.多无人机协同编队飞行控制研究现状及发展[J].哈尔滨工业大学学报,2017,49(3):1-14.  
(Zong Q, Wang D, Shao S, et al. Research status and development of multi UAV coordinated formation flight control[J]. Journal of Harbin Institute of Technology, 2017, 49(3): 1-14.)
- [2] 成成.多无人机协同编队飞行控制关键技术研究[D].北京:中国科学院长春光学精密机械与物理研究所,2018.  
(Cheng C. Research on key technologies of multi-uavs cooperative formation flight control[D]. Beijing: Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, 2018.)
- [3] Tortonesi M, Stefanelli C, Benvegna E, et al. Multiple-UAV coordination and communications in tactical edge networks[J]. IEEE Communications Magazine, 2012, 50(10): 48-55.
- [4] Beard R W. Multiple UAV cooperative search under collision avoidance and limited range-communication constraints[C]. The 42nd IEEE Conference on Decision & Control. Maui, 2004, 1: 25-30.
- [5] Eun Y, Bang H. Cooperative control of multiple unmanned aerial vehicles using the potential field theory[J]. Journal of Aircraft, 2006, 43(6): 1805-1814.
- [6] Ding X C, Rahmani A R, Egerstedt M. Multi-UAV convoy protection: An optimal approach to path planning and coordination[J]. IEEE Transactions on Robotics, 2010, 26(2): 256-268.
- [7] Hayat S, Even Yanmaz, Brown T X, et al. Multi-objective

- UAV path planning for search and rescue[C]. IEEE International Conference on Robotics & Automation. Singapore, 2017: 5569-5574.
- [8] Chen Y, Duan H. Multiple UAVs mission assignment based on modified gravitational search[C]. The 11th IEEE International Conference on Control & Automation (ICCA). Taichung, 2014: 540-545.
- [9] Eun Y, Bang H. Cooperative task assignment/path planning of multiple unmanned aerial vehicles using genetic algorithm[J]. Journal of Aircraft, 2009, 46(1): 338-343.
- [10] 周绍磊, 康宇航, 秦亮, 等. 多无人机协同控制的研究现状与主要挑战[J]. 飞航导弹, 2016(1): 78-83.  
(Zhou S L, Kang Y H, Qin L, et al. Research status and main challenges of collaborative control of multiple UAVs[J]. Aeronautical Missile, 2016(1): 78-83.)
- [11] 施巍松, 孙辉, 曹杰, 等. 边缘计算: 万物互联时代新型计算模型[J]. 计算机研究与发展, 2017, 54(5): 907-924.  
(Shi W S, Sun H, Cao J, et al. Edge computing—An emerging computing model for the internet of everything era[J]. Journal of Computer Research and Development, 2017, 54(5): 907-924.)
- [12] Kim B, Min H, Heo J, et al. Dynamic computation offloading scheme for drone-based surveillance systems[J]. Sensors, 2018, 18(9): 2982.
- [13] Cao X, Jie X, Rui Z. Mobile edge computing for cellular-connected UAV: Computation offloading and trajectory optimization[C]. Signal Processing Advances in Wireless Communications. Kalamata: IEEE, 2018: 1-5.
- [14] Messous M A, Sedjelmaci H, Houari N, et al. Computation offloading game for an UAV network in mobile edge computing[C]. 2017 IEEE International Conference on Communications. Paris, 2017: 1-6
- [15] Valentino R, Jung W S, Ko Y B. Opportunistic computational offloading system for clusters of drones[C]. The 20th International Conference on Advanced Communication Technology (ICACT). Gangwon-do, 2018: 303-306.
- [16] Wang X, Chowdhery A, Chiang M. Networked drone cameras for sports streaming[C]. The 37th International Conference on Distributed Computing Systems (ICDCS). Atlanta, 2017: 308-318.
- [17] Chen X. Decentralized computation offloading game for mobile cloud computing[J]. IEEE Transactions on Parallel and Distributed Systems, 2014, 26(4): 974-983.
- [18] Chen X, Jiao L, Li W, et al. Efficient multi user computation offloading for mobile edge cloud computing[J]. IEEE/ACM Transactions on Networking, 2015, 24(5): 2795-2808.
- [19] Deng M, Tian H, Lyu X. Adaptive sequential offloading game for multi-cell mobile edge computing[C]. The 23rd International Conference on Telecommunications. Thessaloniki, 2016: 1-5.
- [20] Tisdale J, Ryan A, Kim Z, et al. A multiple UAV system for vision-based search and localization[C]. American Control Conference. Seattle, 2008: 1985-1990.
- [21] Gu J, Su T, Wang Q, et al. Multiple moving targets surveillance based on a cooperative network for multi-UAV[J]. IEEE Communications Magazine, 2018, 56(4): 82-89.
- [22] Zhang S, Zeng Y, Zhang R. Cellular-enabled UAV communication: Trajectory optimization under connectivity constraint[J]. 2017, arXiv: 1710.11619.
- [23] Fuhui Z, Yongpeng W, Qingyang H R, et al. Computation rate maximization in UAV-enabled wireless powered mobile-edge computing systems[J]. IEEE Journal on Selected Areas in Communications, 2018, 36(9): 1927-1941.
- [24] Chen M, Miao Y, Hao Y, et al. Narrow band internet of things[J]. IEEE Access, 2017, 5: 20557-20577.
- [25] Monderer D, Shapley L S. Potential games[J]. Games and Economic Behavior, 1996, 14(1): 124-143.

### 作者简介

彭维平(1979—),男,副教授,博士,从事信息安全、物联网安全及应用等研究, E-mail: pwp9999@hpu.edu.cn;

王明坤(1989—),男,硕士,从事智能信息处理与服务的研究, E-mail: 812274207@qq.com;

宋成(1980—),男,副教授,博士,从事信息安全、可信计算等研究, E-mail: songcheng@hpu.edu.cn;

贾宗璞(1963—),男,教授,博士生导师,从事物联网技术及应用、计算机网络技术等研究, E-mail: jiazp@hpu.edu.cn.

(责任编辑: 孙艺红)