

海洋监测数据多级传输控制

常昊天, 冯径, 段超凡, 颜超, 夏凯文

(国防科技大学 气象海洋学院, 江苏, 南京 211101)

摘要: 建立基于海基、空基、天基平台的海空天综合数据传输网络(SASIN)和协同数据传输机制,通过增加具有热点的巡游节点,如无人机、无人艇和船舶等,辅助采集处理浮标/潜标数据,转发至数据中心,从而增加数据整体传输效率,减小传输延时。研究了SASIN中的虚拟机分配和计算任务卸载联合调度控制机制,有效地将计算资源分配给无人机边缘服务器中的不同虚拟机。提出了一种动态网络条件下基于Q学习的计算卸载学习方法,从而处理多维SASIN资源调度。

关键词: 海空天综合数据传输网络; 虚拟机分配; 计算卸载; 联合调度

中图分类号: TP315 **文献标志码:** A **文章编号:** 1001-0645(2020)06-0648-07

DOI: 10.15918/j.tbit.1001-0645.2020.056

Multi-Tier Transmission Control of Marine Observation Data

CHANG Hao-tian, FENG Jing, DUAN Chao-fan, YAN Chao, XIA Kai-wen

(Institute of Meteorology and Oceanography, National University of Defense Technology,
Nanjing, Jiangsu 211101, China)

Abstract: The sea-air-space integrated network (SASIN) and collaborative data transmission mechanism were proposed. By complementing cruising nodes with access points, such as UAVs and unmanned boats, the buoy/submarine buoy data was partially collected, processed and forwarded to the data center, thus increasing the overall data transmission efficiency and reducing the transmission delay. This paper studied the joint scheduling control mechanism of virtual machine allocation, task scheduling and computation task offloading in SASIN. In this paper, a Q-learning based computational offloading method was proposed in dynamic network conditions, so as to deal with the multi-dimension resource scheduling of the SASIN.

Key words: sea-air-space integrated network; virtual machine allocation; computation offloading; joint scheduling

随着海洋探测需求的日益增长,海洋环境监测网络建设在不断发展^[1],海洋环境监测网络属于计算密集型网络应用,对资源受限的终端设备提出了巨大的挑战。虽然云计算的发展有效地满足了计算密集型网络应用需求,但云计算无法满足高实时性要求。为了解决这一问题,移动边缘计算(mobile edge computing, MEC)得到了广泛的研究,它利用

网络边缘的计算资源提供高效、灵活的计算服务。然而,在典型的MEC场景中,边缘计算服务由蜂窝基站或WiFi接入点提供,无法为偏远地区提供无处不在的覆盖。在偏远地区,物联网设备可能会被广泛部署,以执行某些计算要求相对较高的应用,这会迅速耗尽汇聚节点的电量,且造成较大的处理延迟。由于缺乏地面接入网络覆盖,典型的边缘和云

收稿日期: 2020-04-14

基金项目: 国家自然科学基金资助项目(61371119)

作者简介: 常昊天(1990—),男,博士生, E-mail: changhaotian901120@126.com.

通信作者: 冯径(1962—),女,教授,博士生导师, E-mail: jfeng@seu.edu.cn.

计算模式无法应用于此类场景。为此,Cheng 等^[2]建议采用空天地一体化网络(space-air-ground integrated network, SAGIN)架构来减轻远程物联网应用的计算负担,这是一种典型的网络数据多级传输架构。SAGIN 将卫星网络、空中网络与地面网络相结合,提供无缝、灵活的网络覆盖和服务,可应用于智能交通系统、远程监控、灾害救援等领域,以及大规模高速移动互联网接入。SAGIN 是一个由卫星网、空中网和地面网 3 个网段组成的多级异构传输网络,每个网段拥有不同的资源,受到不同的限制。空中网络节点可以作为外围服务器,为 IoT 设备提供低延迟的边缘计算;另一方面,卫星通信虽然可能具有较低的通信速率和较高的传输延迟,但可以通过无缝覆盖和卫星骨干网提供始终在线的云计算。

SAGIN 被认为是解决未来移动通信网络中偏远、远程、大规模覆盖、移动数据增长、数据流量不均、骨干网僵硬等诸多问题的一种可靠技术。SAGIN 的相关研究应用全部集中在地面和空域或天域的边缘层级,未涉及海洋应用背景。在海洋监测网络应用背景下,SAGIN 面临着许多挑战。首先,空中网络的高移动性导致了信道条件和覆盖范围的动态变化;其次,SAGIN 中不同层级的网段具有不同的网络条件和资源约束;最后,SAGIN 中的地面网段模型显然无法应用于海洋背景中。因此,需要对 SAGIN 模型进行修改,且提出适合于解决海洋监测网络数据传输问题的传输机制。本文在基于海空天多网段的海洋监测传输背景下,首次提出了包含边缘计算卸载和计算资源分配的联合数据传输控制机制。

1 海空天综合数据传输网络系统模型

与物联网中使用 SAGIN 相类似,在海洋综合监测网络中对数据进行传输时,为减小网络整体的数据传输延迟,并提高回传数据的实时性和信息价值,也应使用多层级网络结构。因此,本文拟建立海空天平台综合数据传输网络(sea-air-space integrated network, SASIN)和海洋综合监测网络的多级传输控制机制,引入具有移动、通信和计算能力的无人机(unmanned aerial vehicle, UAV)作为边缘计算节点,基于海基、空基和天基 3 个数据传输层级,提高海洋探测数据的传输处理效率。

1.1 SASIN 网络模型

在远程区域部署物联网设备以执行具有计算要求的特定任务,例如监控和视频监控。在所考虑的

偏远地区,没有蜂窝覆盖,因此建立 SASIN,以向物联网设备提供网络接入、边缘计算和缓存等网络功能,SASIN 网络模型如图 1 所示。

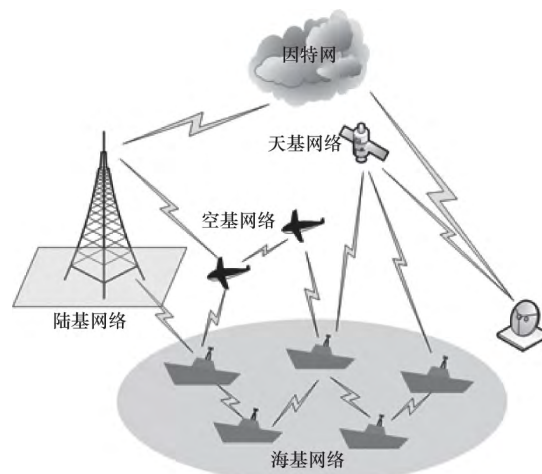


图 1 SASIN 系统模型

Fig. 1 SASIN system model

在 SASIN 网络中,有 3 个网络层次,即海面网段、空中网段和空间网段。SASIN 网络抽象可由图 G 表征为

$$G = (V, U, S, E),$$

式中: $V = \{v_i | i = 1, 2, \dots, N_v\}$ 为浮标节点集合; $U = \{u_j | j = 1, 2, \dots, N_u\}$ 为 UAV 集合; $S = \{s_k | k = 1, 2, \dots, N_s\}$ 为卫星集合; $E = \{(v_i, u_j), (v_i, s_k) | \forall v_i \in V, \forall u_j \in U, \forall s_k \in S\}$ 为浮标与 UAV 间、浮标与卫星间通信链路的集合。

浮标节点构成了海面部分,具有非常有限的能量和计算能力,采集水下传感器的原始探测数据,进行预处理和异常检测标注,按照下文所提出的多级传输控制机制,使用自身、UAV 或者经由卫星使用云端对探测数据进行异常检测处理,最终将有效信息传输至数据中心;在空中部分,飞行的 UAV 可以充当边缘服务器,为地面用户提供边缘缓存和计算能力,UAV 将所处理好的数据信息传回陆基基站,最终传至数据中心;在空间段,低轨卫星(LEO)提供目标区域的全覆盖,并通过卫星骨干网将海面浮标节点与云端相连。

所有浮标节点具有相同的本地计算能力 C^l ,本地任务计算/处理的能量消耗用 ϵ^l 表示, ϵ^l 与 C^l 有关,浮标节点传输到无人机和卫星的功耗分别用 ϵ_i^u 和 ϵ_i^s 表示。在 UAV 边缘服务器中,计算资源虚拟化为虚拟机(virtual machine, VM),每个 VM 用于处理一种特定的海洋要素探测数据。在第 k 个

UAV 中,计算资源总量为 C^e ,第 v 个 VM 的计算资源用 C_v^e 表示,浮标 i 的任务 j 的服务器使用 VM 的成本用 $B_{i,j}^e$ 表示. 对于无人机-海面通信,考虑到任务卸载决策比传统的资源调度时间(毫秒级)要长得多,因此只考虑大规模信道衰减. 无人机和海面浮标之间的路径损耗可由式(1)计算^[3-4]

$$L(r, h) = 20 \lg[4\pi f_c \sqrt{(h^2 + r^2)}] + P_{\text{LoS}}(r, h) \eta_{\text{LoS}} + [1 - P_{\text{LoS}}(r, h)] \eta_{\text{NLoS}}, \quad (1)$$

式中: h 和 r 分别为无人机飞行高度和无人机与海面浮标之间的水平距离; η_{LoS} 和 η_{NLoS} 分别为在 LoS 和 NLoS 链路的自由空间路径损耗的基础上产生的附加损耗; f_c 为载波频率, c 表示光速; P_{LoS} 为无人机-海面链路的视线概率,可以通过式(2)得到

$$P_{\text{LoS}}(r, h) = 1/[1 + ae^{-\text{barctan}(h/r)-a}], \quad (2)$$

式中: $a, b, \eta_{\text{LoS}}, \eta_{\text{NLoS}}$ 为环境相关变量,在偏远地区通常取 4.88, 0.43, 0.1, 21^[5]. 此外,无人机-海面通信使用总带宽为 B 的 WiFi 协议. 每个浮标获得的带宽由式(3)计算为

$$B_i = \rho B \xi(n), \quad (3)$$

式中: ρ 为 WiFi 吞吐量效率因子; $\xi(n)$ 为 WiFi 信道效用函数,是关于浮标数量 n 的递减函数. 因此,即时无人机-海面间的数据速率可以通过式(5)获得

$$r_{\text{SU}} = \rho B \xi(n) \lg(1 + \epsilon_i^e 10^{-L_i/10}/\sigma^2), \quad (4)$$

式中: L_i 为相应浮标至 UAV 链路的路径损耗; σ^2 为高斯噪声的功率. 卫星-海面通信数据速率 r_{SS} 小于无人机-海面数据速率, r_{SC} 表示卫星和云之间的传输速率. 云计算能力远高于浮标和 UAV 边缘服务器,每个任务的处理速率用 C^e 表示,云端处理浮标 i 上数据 j 的成本用 $B_{i,j}^e$ 表示.

1.2 计算卸载模型

在 SASIN 网络中,有 M 个浮标和 N 种不同的海洋要素探测数据,每个浮标处理 N 种不同的海洋要素数据. 不同海洋要素数据具有一定的优先级,要素序号较小的数据将比序号较大的更早地被发送/处理. 对于数据类型 j ,输入数据和工作负载分别用 H_j 和 Z_j 表示.

然而,由于浮标的能量和计算能力有限,计算任务也可以被卸载到无人机边缘服务器或通过卫星被卸载到云上,在每个时隙进行卸载决策,直到完成所有 $M \times N$ 任务. 在时隙 t 的开始处,剩余的任务用 $M \times N$ 矩阵 $\mathbf{M}(t)$ 表示,其中元素 $m_{ij}(t) = 1$ 表示任务 W_{ij} 尚未完成,否则 $m_{ij}(t) = 0$. 分别用 $M \times N$ 矩

阵 $\mathbf{X}_l(t)$, $\mathbf{X}_e(t)$ 和 $\mathbf{X}_c(t)$ 表示在时隙 t 上本地处理任务、将任务卸载到边缘和将任务卸载到云的决策,并且每个二进制元素 $x_{ij}^l(t)$, $x_{ij}^e(t)$ 和 $x_{ij}^c(t)$ 表示任务 W_{ij} 分别是在本地处理、卸载到边缘还是卸载到云. 请注意,任务 W_{ij} 在时间 t 最多只可以调度到一个设备上,即卸载决策受式(5)(6)约束为

$$x_{ij}^l(t), x_{ij}^e(t), x_{ij}^c(t) \in \{0, 1\}, \quad (5)$$

$$x_{ij}^l(t) + x_{ij}^e(t) + x_{ij}^c(t) \leq m_{ij}(t). \quad (6)$$

当一个未完成的任务没有在时隙 t 被调度时式(6)中的不等式成立. 如果任务 W_{ij} 在时间 t 被本地处理或卸载到云上,认为任务可以在一定的延迟下完成,并且 $m_{ij}(t) = 0$. 但是,如果 W_{ij} 卸载到 UAV 边缘服务器,它可能无法完成并在 t 结束时完成,主要是因为如果将多个任务卸载到一个 UAV 边缘服务器,其中一些任务可能无法在时间段内完成.

1.3 成本模型

计算卸载旨在最小化系统成本,系统成本包括延迟成本及能耗和服务器使用成本.

① 延迟成本:如果任务 W_{ij} 被安排在时隙 t ,则可以根据卸载决策计算延迟. 如果任务计划在本地处理,则延迟为

$$T_{ij}^l = \epsilon(t-1) + t_{r,i}^l + Z_j/C^l, \quad (7)$$

式中: ϵ 为时隙的长度, $\epsilon(t-1)$ 为自任务生成以来经过的时间,由于浮标的计算能力较低,在时隙 t 初始,可能有一些任务滞留在本地; $t_{r,i}^l$ 为浮标 i 完成本地剩余任务所需的时间,由剩余负载除以本地算力 C^l 来计算. 如果将任务卸载至边缘,并在时隙 t 内完成,则延迟为

$$T_{ij}^e = \epsilon(t-1) + d_{ij}^e + \frac{1}{r_{\text{SU}}} \sum_{a=1}^j x_{i,a}^e(t) H_a, \quad (8)$$

式中: d_{ij}^e 为 W_{ij} 在边缘的处理延迟,取决于卸载决策和 VM 资源分配. 任务卸载到云端的延迟为

$$T_{ij}^c = \epsilon t + Z_j/C^c + H_j \left(\frac{1}{r_{\text{SC}}} + \frac{1}{r_{\text{SS}}} \right). \quad (9)$$

② 能源和服务器使用成本:本地处理 W_{ij} 的能源成本可以通过式(10)计算

$$L_{ij}^l = \epsilon^l Z_j/C^l. \quad (10)$$

如果在时隙 t ,任务 W_{ij} 被卸载到 UAV,能量和服务器使用成本可以通过式(11)得到

$$L_{ij}^e = \epsilon_i^e \sum_{\tau=1}^t x_{i,a}^e(\tau) \frac{1}{r_{\text{SU}}(\tau)} H_j + \alpha B_{ij}^e, \quad (11)$$

式中 α 为 UAV 使用成本相对于传输能耗的权重. 类似地,云上能耗和服务器成本为

$$L_{ij}^s = \epsilon_j H_j / r_{sc} + \beta B_{ij}^s. \quad (12)$$

1.4 数据优先级定义

本文提出海洋综合监测网络多级传输中的数据优先级概念,并对数据优先级概念进行定义. 回顾信息价值的定义,为了对目标区域的及时探索可为采取对策提供充分的准备时间,采用信息价值(value of information, VoI)衡量传感数据的重要敏感程度,当水下数据包 p_i^j 在 t 时刻被传递到浮标节点 i 时,其信息价值 $f_{\text{Vol}}(p_i^j)$ 被浮标节点 i 作为初始信息价值.

同时, p_i^j 对应于时刻 t 数据类型 j ,可并入 W_{ij} ,可进一步进行局部数据监测,得到 $P(p_i^j) = P(p_i^j | p_{i-1}^j, p_{i-2}^j, \dots, p_{i-k}^j)$,则信息价值记为 $\xi_p = f_{\text{Vol}}(p_i^j) / P(p_i^j)$.

2 多优先级数据传输控制机制

2.1 虚拟机计算资源分配机制

在时隙 t 中,可以将多个任务卸载到一个边缘服务器的多个 VM 中并行执行,以减少处理延迟. 一个 VM 执行特定要素数类型任务. 因此,本节研究虚拟机分配问题,选择边缘承接的任务以及将计算资源分配给不同的虚拟机. 由于边缘服务器的总计算资源是固定的,因此需共同优化无人机边缘服务器中的虚拟机分配和任务调度,以减少系统总延迟.

假设有多种要素探测数据 $\mathcal{A} = \{1, 2, \dots, N\}$, 和一个计算能力为 C^e cycles/s 的边缘服务器. 用 $\mathcal{T}_m = \{1, 2, \dots, N_m\}$ 表示第 m 种要素数据的各个等量计算任务,但最大延迟要求不同. Z_m 表示第 m 个要素数据任务的计算工作量, $C = \{c_m | m \in \mathcal{A}\}$, c_m 表示 UAV 上第 m 个 VM 的计算资源变量, $Y = \{y_{m,n} | m \in \mathcal{A}, n \in \mathcal{T}_m\}$ 表示决策变量,其中 $y_{m,n} = \{0, 1\}$, 若类型 m 任务 n 被调度和执行,则 $y_{m,n} = 1$. 定义最小化总延迟问题如下

$$\min_{C, Y} \sum_{m=1}^N \sum_{n=1}^{N_m} \left[y_{m,n} \sum_{k=1}^n y_{m,k} Z_m / c_m + \epsilon (1 - y_{m,n}) \right],$$

s. t.

$$\begin{cases} C_1: \sum_{k=1}^n y_{m,k} Z_m / c_m \leq t_{m,n}, \forall m \in \mathcal{A}, \forall n \in \mathcal{T}_m, \\ C_2: \sum_{m=1}^N c_m \leq C^e, \\ C_3: c_m \geq 0, \\ C_4: y_{m,n} \in \{0, 1\}, \forall m \in \mathcal{A}, \forall n \in \mathcal{T}_m. \end{cases}$$

式中 $t_{m,n}$ 为数据 m 的任务 n 的延迟要求, $t_{m,n}$ 可以通

过式(13)得到

$$t_{m,n} = \epsilon / \xi_{m,n}, \quad (13)$$

式中 $\xi_{m,n}$ 为任务 W_{ij} 的信息价值.

由此可见,这是一个难以求解的混合整数规划问题. 它包括虚拟机计算资源向量 C 和 $0 \sim 1$ 整数矩阵 Y . 即使假设 C 是已知的,剩余子问题仍然是一个具有 $0 \sim 1$ 整数约束的二次问题,这对于非正定矩阵是 NP 困难的. 该问题通常用特定的松弛方法重新构造,然后用凸优化技术求解,这种非在线方法需要大量的迭代,实时性较差. 因此,本节设计一个有效的低复杂度的虚拟机资源分配算法来获得次优解,如图 2 所示.

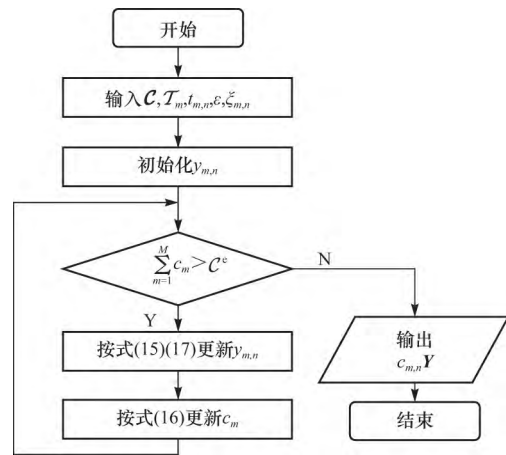


图2 虚拟机资源分配算法

Fig. 2 Resource allocation algorithm of VM

在所提出的虚拟机分配和任务调度算法中,假设虚拟机 m 的任务延迟要求按序排列,即 $t_{m,n} \leq t_{m,n+1}$. 初始化 c_m 且令 $y_{m,n} = 1, \forall m \in \mathcal{A}, \forall n \in \mathcal{T}_m$, $c_m = \min\{(nZ_m)/t_{m,n}\}$, (14)

如果 $\sum_{m=1}^N c_m > C^e$, 则意味着并非所有任务都可以调度,舍弃具有最苛刻延迟要求的任务

$$y_{m',n'} = 0, \quad (m', n') = \min_{m,n} t_{m,n}, \quad (15)$$

然后,按式(16)再次为 VM 分配 c_m ,重复直到满足约束条件,并且获得 C 和任务调度 Y ,

$$c_m = \min\left\{\left(\sum_n y_{m,n} Z_m / t_{m,n}\right), \forall m \in \mathcal{A}, \forall n \in \mathcal{T}_m\right\}, \quad (16)$$

非计划任务则由式(17)选出

$$(m', n') = \arg \max_{m,n} \left\{ \sum_n y_{m,n} Z_m / t_{m,n} \right\}. \quad (17)$$

最坏情况需要 $N'(N'+1)/2$ 次比较,其中 N' 是边缘服务器的总卸载任务数.

2.2 计算卸载调度机制

为 SASIN 网络系统设计一种在线计算卸载方法, 计算任务被安排在本地处理, 卸载到 UAV 边缘服务器, 或通过卫星卸载到云服务器, 以最小化任务延迟、能耗以及服务器的使用成本. 将计算卸载决策建模为 Q-learning 过程, 定义为结构体 $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R})$, 其中 \mathcal{S} 代表系统状态集合, \mathcal{A} 代表动作集合, \mathcal{T} 代表状态转移概率, \mathcal{R} 则代表在系统某状态 $s \in \mathcal{S}$ 下采取动作 $a \in \mathcal{A}$ 的奖励或代价函数. SASIN 计算卸载问题的 Q-learning 模型可描述为 4 方面:

① 状态(state): 在时隙 t 的初始, 系统状态 $S(t) = M(t) \otimes T_r^1(t)$, $M(t)$ 和 $T_r^1(t)$ 分别为 t 时隙浮标上的剩余任务标志矩阵和任务剩余处理时间矩阵.

② 动作(action): 为任务分配卸载计划, 即 $a(t) = \{X_l(t), X_e(t), X_c(t)\}$.

③ 状态转移概率(transition probability): 表明本时隙到下一时隙状态的转移概率

$$P\{S(t+1) | S(t), a(t)\} = P\{M(t+1) \otimes T_r^1(t+1) | M(t) \otimes T_r^1(t), a(t)\},$$

剩余处理时间矩阵中的对应元素

$$t_{r,i}^1(t+1) = \max\{t_{r,i}^1(t) + \sum_j x_{ij}^1(t) Z_j / C^l - \epsilon, 0\},$$

这样一来状态转移概率受到 $\sum_j x_{ij}^1(t) Z_j / C^l$ 影响, 也即受 $X_l(t)$ 影响.

④ Q 值(Q-value): Q-learning 的难点在于 Q 值的建立. Q 值即为自初始状态起、以动作-状态映射的系统价值量, 可定义为

$$Q(S, a) = E\left[\sum_{t=0}^{\infty} \gamma C(S(t), a(t))\right],$$

式中 $\gamma \in [0, 1]$ 为折算因子, 所有状态轨迹所对应的代价函数和的期望值即为 Q 值.

代价函数由 3 部分组成, 分别是加权任务延迟、能耗和服务器成本. $C(S(t), a(t)) = \sum_{i,j} C_{ij}(s_{ij}(t), a_{ij}(t))$ 为任务 W_{ij} 在时隙 t 的代价函数.

I) 当 $m_{ij} = 0$ 时, 任务已经完成, 所以 $C_{ij}(s_{ij}(t), a_{ij}(t)) = 0$;

II) 当 $m_{ij} = 1$, $x_{ij}^l(t) + x_{ij}^e(t) + x_{ij}^c(t) = 0$ 时, $C_{ij}(s_{ij}(t), a_{ij}(t)) = \epsilon$;

III) 当 $m_{ij} = 1$ 且 $x_{ij}^l(t) + x_{ij}^e(t) + x_{ij}^c(t) = 1$ 时, 有新任务指派, $C_{ij}(s_{ij}(t), a_{ij}(t))$ 为

$$x_{ij}^l(t)(T_{ij}^l - \epsilon t) + x_{ij}^e(t)(T_{ij}^e - \epsilon t) + x_{ij}^c(t)(T_{ij}^c - \epsilon t) + x_{ij}^l(t)L_{ij}^l + x_{ij}^e(t)L_{ij}^e + x_{ij}^c(t)L_{ij}^c.$$

本节所要实现的就是提出一种寻找最优动作的在线方法, 如式(18)所示

$$a^* = \arg \min_{a \in \mathcal{A}} \{C(S, a) + \gamma Q(S, a)\}. \quad (18)$$

由于浮标无法知道未来自身、UAV 和卫星的通信状况以及任务卸载状况, 因此必须采用在线算法计算资源分配和任务卸载状况, 利用以往的信息预测未来的状态转移规律

$$Q(s_{ij}(t), a_{ij}(t)) = C(s_{ij}(t), a_{ij}(t)) + \gamma \sum_{a_{ij}(t+1)} P\{a_{ij}(t+1)\} C(s_{ij}(t+1), a_{ij}(t+1)). \quad (19)$$

$Q(s_{ij}(t), a_{ij}(t))$ 在 t 时刻初始化为代价函数 $C(s_{ij}(t), a_{ij}(t))$, 而 $C(s_{ij}(t+1), a_{ij}(t+1))$ 按 t 时刻的状态信息得到, 之后按式(19)更新, 概率 $P\{a_{ij}(t+1)\}$ 由以往的统计频数得到

$$P\{a_{ij}(t)\} = x_{ij}^l(t)P\{x_{ij}^l(\tau+1) | x_{ij}^l(\tau)\} + x_{ij}^e(t)P\{x_{ij}^e(\tau+1) | x_{ij}^e(\tau)\} + x_{ij}^c(t)P\{x_{ij}^c(\tau+1) | x_{ij}^c(\tau)\}. \quad (20)$$

对应不同动作 $a(t) = \{a_{ij}(t)\}$ 计算 Q 值, 挑选出具有最小 Q 值的动作组合, 形成决策方案, 计算卸载调度算法流程如图 3 所示, 图中 $a^*(t)$ 计算方式如式(21)

$$a^*(t) = \arg \min_{a(t) \in \mathcal{A}} \left\{ \sum_{i,j} Q(s_{ij}(t), a_{ij}(t)) \right\}. \quad (21)$$

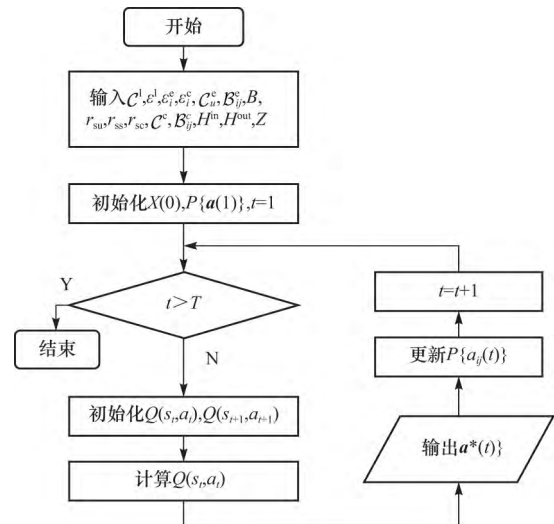


图 3 计算卸载调度算法

Fig. 3 Scheduling algorithm of computation offloading

3 仿真实验

3.1 仿真参数设置

评估了针对无人机边缘服务器提出的联合虚拟

机资源分配和任务调度方案,以及针对 SASIN 网络系统的基于增强学习的在线计算卸载方法。考虑远程 $1\text{ km} \times 1\text{ km} \times 1\text{ km}$ 的立体区域,30 个浮标固定在该区域。海洋传感器采集 5 种不同的要素数据。浮标计算能力 C^l 设为 200 MC/s ,本地任务处理的能耗为 140 mW 。浮标向无人机和卫星的发送和接收功率为 200 mW ,使用 5 架无人机作为飞行边缘服务器。无人机运动轨迹的规划沿用了 Wu 等^[6]的工作,最大限度地提高最小吞吐量。一个 LEO 卫星为海面探测区域进行全覆盖,卫星地面通信速率 r_{ss} 和卫星云数据速率 r_{sc} 都设置为 10 Mb/s ;计算与数据比率为 1 000,即 $Z=1\text{ 000 } H$, H 大小为 5 MB , $B_{ij}^c = B_{ij}^s = W_{ij}$ 。此外,本文工作中,使用 Intel(R) Core(TM) i7-7700HQ CPU@ 2.80 GHz , 8 GB RAM 计算机,基于 Matlab 2014a 进行仿真模拟实验。

3.2 虚拟资源分配

将启发式虚拟机计算资源分配和任务调度算法与暴力方法和随机方法进行比较。图 4 示出了所提出算法的延迟性能,相对于边缘服务器计算资源的平均延迟。图 5 表示了当边缘计算速度为 10 GC 时平均延迟与卸载任务总数之间的关系。在图 4 与图 5 中,所提启发式算法都能达到与暴力方法非常接近的性能。

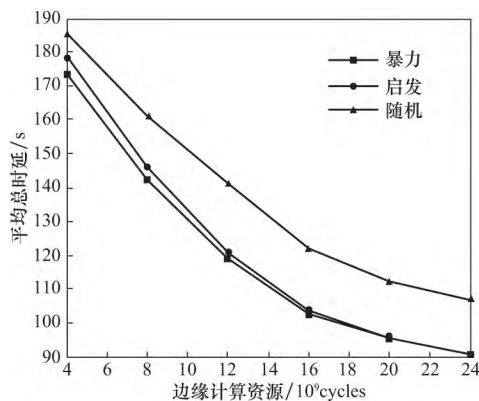


图 4 平均总时延—边缘计算资源

Fig. 4 Average sum delay vs. MEC resources

如图 6 所示,随着任务总数的增加,暴力穷举方法的运行时间呈指数增长,而启发式算法的运行时间相对小得多。所提虚拟机计算资源分配和任务调度算法能够同时达到接近最优的性能和较低的计算复杂度,能够在动态网络条件下为边缘服务器进行有效资源分配。

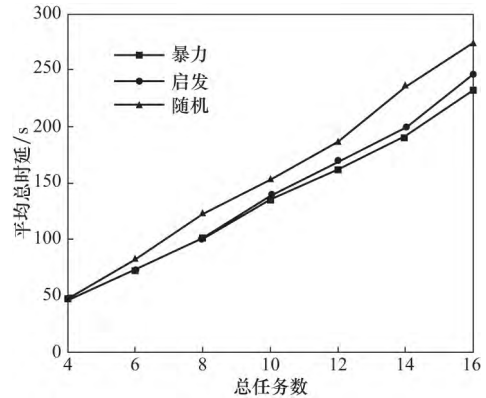


图 5 平均总时延—总任务数

Fig. 5 Average sum delay vs. total number of tasks

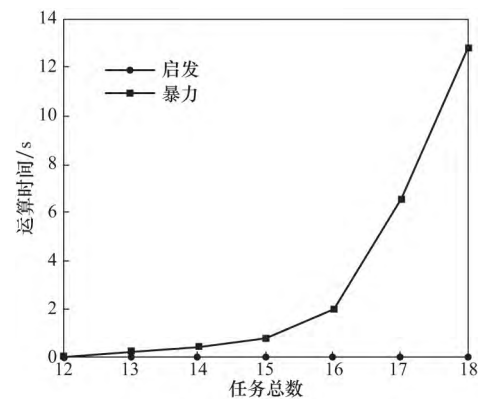


图 6 运算时间比较

Fig. 6 Comparison of computing time

3.3 计算卸载调度

本小节对提出的计算卸载方法进行了评估和比较,将其与另外两种计算卸载方法,即随机和贪婪进行了比较。随机方法随机卸载任务;而贪婪法要求浮标向通信范围内的全部 UAV 卸载任务,否则依固定概率进行本地或云卸载。图 7 刻画了本文所提卸载算法的收敛性能,从第 12 个时隙左右算法已经收敛,所构建的成本函数增强了算法的学习。

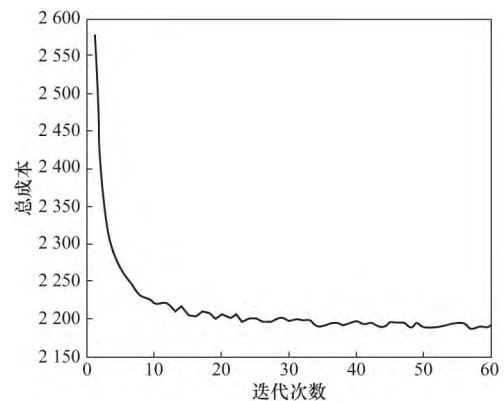


图 7 算法收敛性能

Fig. 7 Convergence of the algorithm

表 1 展示了成本的主要组成部分,即能耗和延迟. 本文所提出的计算卸载方法可以实现最低的能耗和最低的延迟. 随机方法获得与基于 Q 学习的方案相近能耗,因为随机方法在本地处理更多任务,因此能耗较低,但由于本地延迟较长,因此延迟更高.

表 1 能耗延时比较

Tab. 1 The comparison of energy consumption and delay

方法	能耗/J	延时/s
启发	1 760	7 867
随机	1 810	13 625
贪婪	7 912	17 982

然而,贪婪方法具有很高的能量消耗和延迟,这是由于向无人机边缘服务器上传了过多任务,增大了传输能耗,且增加了处理延迟. 这可以在表 2 中得证,所提出的方法比其他两种卸载方式更频繁地选择卫星云. 与卫星云相比,边缘服务器虽然具有高传输速率,但面临竞争问题;而本地处理由于计算能力弱,则会导致较长的延迟. 贪婪方法选择更多的无人机边缘,而随机方法则较为均匀地选择了 3 种方式.

表 2 卸载分配情况

Tab. 2 The offloading allocation

方法	本地	边缘	云
启发	32	8	110
随机	70	6	74
贪婪	48	49	53

图 8 表示了卸载性能相对于 UAV 服务器使用成本权重 α 的关系,基于 Q 学习的方法通过与环境的交互获取卸载策略,实现最低的总成本. 总成本关于云服务器使用成本权重 β 的变化趋势如图 9 所示,基于 Q 学习的卸载方法实现了最低的总成本,但总成本增长最快.

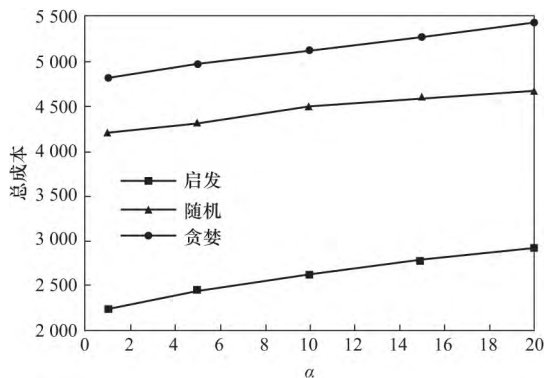


图 8 总成本随 α 的变化

Fig. 8 Total cost vs. parameter α

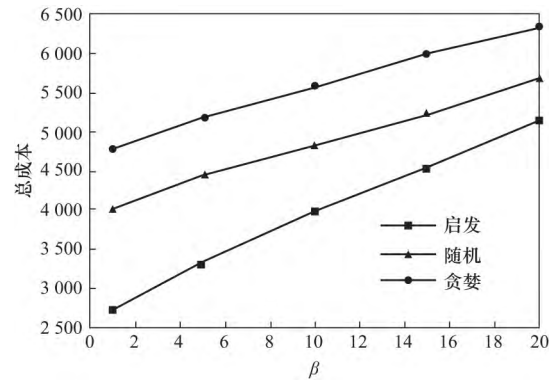


图 9 总成本随 β 的变化

Fig. 9 Total cost vs. parameter β

4 小 结

本文构建了 SASIN 系统模型,研究了 SASIN 中的虚拟机分配和任务调度和计算任务卸载联合调度控制机制,有效地将计算资源分配给无人机边缘服务器中的不同虚拟机,同时减轻了计算密集型任务的负担. 提出了一种动态条件下基于 Q 学习的计算卸载学习方法,从而处理多维 SASIN 资源调度,仿真结果验证了所提方法的收敛性和有效性.

参考文献:

- [1] Chang H, Feng J, Duan C. Reinforcement learning-based data forwarding in underwater wireless sensor networks with passive mobility[J]. Sensors, 2019, 19(2): 256–265.
- [2] Cheng N, Lyu F, Quan W, et al. Space/aerial-assisted computing offloading for IoT applications: a learning-based approach[J]. IEEE Journal on Selected Areas in Communications, 2019, 37(5): 1117–1129.
- [3] Shi W, Li J, Xu W, et al. Multiple drone-cell deployment analyses and optimization in drone assisted radio access networks[J]. IEEE Access, 2018; 1.
- [4] Al-Hourani A, Kandeepan S, Lardner S. Optimal LAP altitude for maximum coverage[J]. Wireless Communications Letters, IEEE, 2014, 3(6): 569–572.
- [5] Boryaliniz R I, Elkeyi A, Yanikomeroğlu H, et al. Efficient 3-D placement of an aerial base station in next generation cellular networks[C] // Proceedings of International Conference on Communications. [S. l.]: IEEE, 2016: 1–5.
- [6] Yang D, Wu Q, Zeng Y, et al. Energy trade-off in ground-to-UAV communication via trajectory design [J]. IEEE Transactions on Vehicular Technology, 2018; 1.

(责任编辑:李兵)