

低轨卫星网络中业务图驱动的星间协作计算方案

国晓博¹, 任智源², 程文驰², 纪 哲³

(1.中国电子科技集团第五十四研究所, 河北 石家庄 050081;

2.西安电子科技大学综合业务网理论及关键技术国家重点实验室, 陕西 西安 710071;

3.北京邮电大学信息与通信工程学院, 北京 100876)

摘 要: 低轨卫星网络具备覆盖范围广、星地链路短等优点, 成为当前的热点研究领域。针对将业务卸载到云计算中心产生过高传输时延的问题, 提出在低轨卫星网络中实现业务图驱动的星间协作计算方法。该方法首先依托加权时间扩展图模型, 屏蔽低轨卫星网络的动态性; 之后联合多颗卫星的计算资源和星间链路资源, 将有向无环图形式的业务调度到传输路径内的卫星上, 实现业务在低轨卫星网络内的“边传输边计算”。仿真结果表明, 当数据量取5 MB时, 星间协同计算的时延性能相比地面云计算提高了58.9%。

关键词: 低轨卫星网络; 时间扩展图; 星间协作计算

中图分类号: TN927+.23

文献标识码: A

doi: 10.11959/j.issn.2096-8930.2021018

Inter-Satellite Cooperative Computing Scheme Driven by Business Graph in LEO Satellite Network

GUO Xiaobo¹, REN Zhiyuan², CHENG Wenchi², JI Zhe³

1. The 54th Research Institute of China Electronics Technology Group Corporation, Shijiazhuang 050081, China

2. State Key Laboratory of Integrated Services Networks, XiDian University, Xi'an 710071, China

3. School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract: Low earth orbit-satellite network (LEO-SN) has the advantages of wide coverage and short satellite-to-earth link, and has become a current hot research field. A business graph-driven inter-satellite collaborative computing method in LEO-SN was proposed to solve the problem of high transmission delay when the task was offloaded to the cloud computing center. This method shielded the dynamics of LEO-SN by relying on the weighted time expansion graph model, combined multiple satellite computing resources, and dispatches tasks in the form of a directed acyclic graph to the satellites in the transmission path, thereby realizing the "computing while transmitting" of user services in the satellite cluster. The simulation results showed that when the data volume was 5 MB, the latency performance of on-orbit collaborative computing was improved by 58.9% compared to ground cloud computing.

Keywords: low earth orbit-satellite network, time expanded graph, inter-satellite cooperative computing

1 引言

卫星网络具备覆盖范围广、抗毁性强等显著优点, 目前已被应用到应急通信、导航定位等领域中^[1]。特别地, 低轨卫星网络 (Low Earth Orbit-Satellite Network, LEO-SN) 凭借星地传输距离短、建造成本低等独特优势获得了学术界和工业界的广泛关注^[2]。

现有的LEO-SN^[3]均采用弯管式传输数据, 海量用户数据通过LEO-SN转发至地面云中心, 云平台凭借其强大的计算能力为用户提供集中式的计算服务。卫星转发的数据传到地面云中心产生的通信开销大、时延高, 难以有效支撑抗震救灾等时延敏感型业务。因此, 卫星在轨计算得到广大学者的关注, 通过将任务直接卸载在

收稿日期: 2020-12-23; 修回日期: 2021-03-07

通信作者: 任智源, zyren@xidian.edu.cn

基金项目: 科技部重点研发计划资助项目 (No.2019YFB1803305)

Foundation Item: Key Research and Development Plan of Ministry of Science and Technology (No. 2019YFB1803305)

卫星上进行计算可显著降低传输时延^[4-6]。受限于卫星载荷能力和宇宙射线的制约^[7],低轨(Low Earth Orbit, LEO)卫星单星计算能力难以得到大幅度提升,将业务卸载在单颗LEO卫星上将导致计算时延过高,同样无法满足业务的低时延需求。因此,利用整个LEO-SN的计算资源,进行多星协同计算具有广阔的发展前景。和云计算模式相比,可避免较远的星地链路传输所带来的高传输时延问题;和单星在轨计算方案相比,可解决因单星计算能力不足所导致的高计算时延问题。基于此,针对链路高时延、单星计算能力弱的LEO-SN环境亟须研究一种多星协同计算技术,以满足日益增长的业务计算需求。

利用多星进行协同计算仍面临许多问题。首先,卫星的高动态特性导致其网络拓扑结构在时刻变化,链路连接关系的不确定性给多星协同计算带来了很大的挑战;其次,多颗LEO卫星之间如何协同,以充分利用计算资源,高效地对任务进行调度,进而满足业务需求也极具挑战。

针对以上问题,本文首先依托加权时间扩展图(Weighted Time Expanded Graph, WTEG)模型来屏蔽LEO-SN的高动态特性。该模型将LEO-SN拓扑在时间上离散、空间上延展,构成一张包含完整网络信息的稳态图,并为该稳态图的边增加统一的时延权重参数来分析业务在轨计算和传输产生的时延;之后联合多颗卫星资源计算,将地面通信网络中已有的一种业务图驱动的路径式协同计算技术^[13]引入LEO-SN。该技术将有向无环图形式的任务调度到传输路径内的卫星上,从而实现用户业务在卫星集群内的“边传输边计算”。

2 相关工作

关于“协同计算技术”的研究,地面移动通信网络出现了许多利用分布在用户“边缘”的网络设备集群进行协同计算的技术,主要有基于数据分割的协同计算技术^[8]、基于负载均衡的协同计算技术^[9]、业务图驱动的协同计算技术^[10-13]等。其中业务图驱动的协同计算技术顺应业务复杂化趋势,采用有向无环图(Directed Acyclic Graph, DAG)模型表征业务,通过指定业务图到网络拓扑的映射实现业务调度,单个网络设备只需加载映射到其上的业务图节点所需的服务功能,即能有效降低单个网络设备的计算负载,适用于载荷能力受限的LEO-SN。早期,为了优化网络的业务传输速率,相关学者研究了不同业务图模型映射到网络拓扑上的最大传输速率,但是上述研究没有考虑节点计算能力的约束^[10]。随后,为了联合优化业务计算和传输的总体性能,参考文献[11]提出了业务图的边和节点联合映射到网络拓扑的技

术,将节点的计算能力约束考虑在内,但是其不适用于具有分支的业务图模型。针对此问题,参考文献[12]制定了业务图可分支的映射优化方案,将业务图适用范围扩展到任意的有向无环图模型。此外,业务图驱动的协同计算技术还被引入物联网^[13]等应用场景。但在LEO-SN场景与业务图驱动的协同计算技术融合方面,暂未见到公开发表的论文与技术成果。

关于LEO-SN动态性问题的研究,很多学者提出了时间扩展图模型,将LEO-SN拓扑在时间上离散、空间上延展,构成一张包含完整网络信息的高维稳态图。参考文献[14]利用时间扩展图模型表征LEO-SN,分析卫星收发器资源的最佳分配技术。参考文献[15]通过修改时间扩展图模型将低轨、高轨及地面网络资源整合到一起,提出了优化资源配置的策略。但上述研究中卫星仅作为交换设备,转发业务数据,时间扩展图模型缺少时间相关参数,不能用于在轨协同计算的时延性能分析。

因此,针对上述现存问题,本文基于WTEG模型展开研究,为时间扩展图的边增加统一的时延权重参数,用于分析业务在轨计算和传输产生的时延;并在此基础上,将业务图驱动的协同计算技术引入LEO-SN场景,达到在轨协同计算的目的。

3 基于加权时间扩展图的卫星动态系统建模

为了能够利用LEO-SN计算资源实现在轨协同计算,本文基于典型的空基云雾网络架构^[16]展开研究,如图1所示。

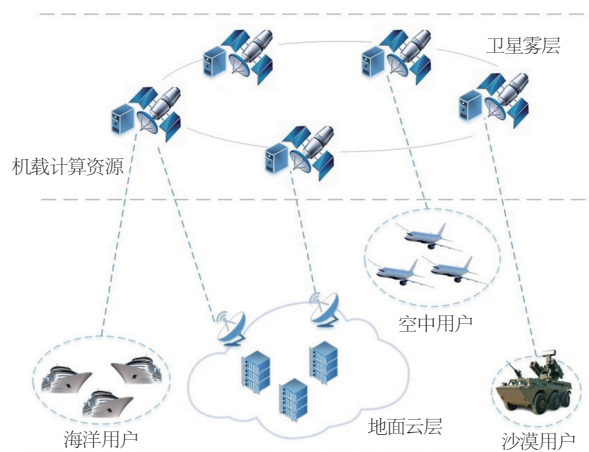


图1 空基云雾网络架构

该架构分为卫星雾层和地面云层两部分。卫星雾层由具有一定计算能力的LEO卫星集群组成,能够为地面用户提供计算服务。为保证服务质量,当业务请求到

达发起卫星后,业务数据被卸载到传输路径中的卫星上进行协同计算,在传输过程中完成业务处理,将处理结果直接传给目标用户;地面云层由高性能服务器组成,通过地面站与卫星雾层相连,对卫星雾层进行监控和管理。此外,为了实现在轨协同计算,地面云层需要根据卫星网络状态、业务图模型等信息,预先规划业务调度技术,并将调度指令缓存到相应的执行卫星。

3.1 卫星运行轨道参数约束下的时间扩展图

为屏蔽LEO-SN的动态性,本小节通过LEO卫星运行轨道参数构建时间扩展图模型,将LEO-SN动态拓扑转换成稳态图模型。

为了便于轨迹建模,本小节假设LEO卫星以地心为圆心,围绕地球表面做周期性圆周运动。因此,本小节以地心为坐标原点,以赤道平面为 X 轴、 Y 轴所在平面, Z 轴通过地心并垂直于赤道平面指向北极,建立空间直角坐标系。此时,LEO-SN可以表示为LEO卫星的集合 $N=\{v_1, \dots, v_p, \dots, v_d\}$,其中 d 为低轨卫星数目;同时,对于任意卫星 $v_p \in N$ 可以用4个参数描述其轨道,即 $v_p \triangleq \{R_p, \alpha_p, \beta_p, \theta_p\}$;其中 R_p 为 v_p 的轨道半径, α_p 为 v_p 的轨道倾角, β_p 为 v_p 的轨道与赤道平面的交线与 Y 轴正半轴的夹角, θ_p 为 v_p 的初始相位,如图2所示。

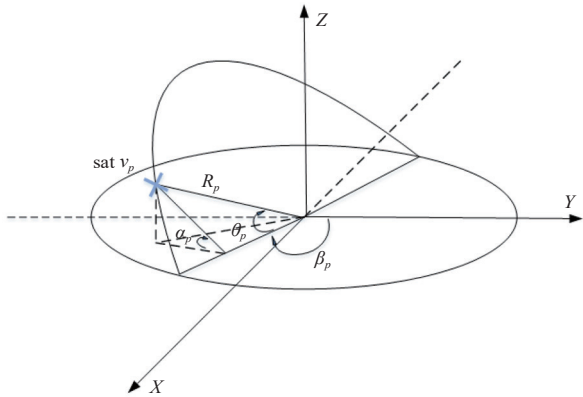


图2 低轨卫星坐标

则在任意时刻 τ , v_p 的坐标可以表示为式(1)

$$\begin{cases} x_p(\tau) = R_p \cos(\omega\tau + \theta_p) \sin(\beta_p) + R_p \sin(\omega\tau + \theta_p) \cos(\alpha_p) \cos(\beta_p) \\ y_p(\tau) = R_p \cos(\omega\tau + \theta_p) \cos(\beta_p) - R_p \sin(\omega\tau + \theta_p) \cos(\alpha_p) \sin(\beta_p) \\ z_p(\tau) = R_p \sin(\omega\tau + \theta_p) \sin(\alpha_p) \end{cases} \quad (1)$$

在时刻 τ , v_p 与 $\{v_o | v_o \in N, p \neq o\}$ 之间的距离如式(2)所示

$$\text{dis}_{p_o}(\tau) = \sqrt{(x_p(\tau) - x_o(\tau))^2 + (y_p(\tau) - y_o(\tau))^2 + (z_p(\tau) - z_o(\tau))^2} \quad (2)$$

定义 τ 时刻 v_p 与 v_o 之间的链路状态如式(3)所示,当 $\text{link}_{p_o}(\tau) = r^*$ 表示 v_p 与 v_o 连通且链路传输速率为 r^* ;反之, v_p 与 v_o 链路中断

$$\text{link}_{p_o}(\tau) = \begin{cases} r^*, & r_{p_o}(\tau) \geq r^* \\ 0, & r_{p_o}(\tau) < r^* \end{cases} \quad (3)$$

其中 r^* 表示星间链路的设计速率, $r_{p_o}(\tau)$ 为 τ 时刻 v_p 与 v_o 的理论传输速率

$$r_{p_o}(\tau) = B_s \text{lb} \left\{ 1 + \frac{P_r(\tau)}{\sigma^2 B_s} \right\} \quad (4)$$

其中, B_s 为星间链路带宽, σ^2 为高斯白噪声方差, $P_r(\tau)$ 为时刻 τ 的信号接收功率。考虑到自由空间中路径损耗在星间链路信号传输损耗中占主导地位^[20],因此, $P_r(\tau)$ 可由式(5)表示

$$P_r(\tau) = \frac{P_t G_t G_r}{(4\pi \text{dis}_{p_o}(\tau) / \lambda)^2} \quad (5)$$

与 $P_r(\tau)$ 同理, G_r 、 P_t 、 G_t 分别表示信号接收增益系数、信号发射功率和信号发射增益系数, λ 为载波波长。

由式(4)和式(5)可知,式(3)可改写为式(6)

$$\text{link}_{p_o}(\tau) = \begin{cases} r^*, & B_s \text{lb} \left\{ 1 + \frac{P_t G_t G_r}{(4\pi \text{dis}_{p_o}(\tau) / \lambda)^2 \sigma^2 B_s} \right\} \geq r^* \\ 0, & B_s \text{lb} \left\{ 1 + \frac{P_t G_t G_r}{(4\pi \text{dis}_{p_o}(\tau) / \lambda)^2 \sigma^2 B_s} \right\} < r^* \end{cases} \quad (6)$$

通过式(6)可计算出两颗卫星任意时刻的链路状态,则LEO-SN动态拓扑可通过遍历所有卫星对得到。为将LEO-SN的动态拓扑稳态化,本文基于时间扩展图来屏蔽动态性。首先以LEO-SN某一时刻的状态为起点,将系统运行周期 T 等分为 n 个连续且大小相等的时隙 $\{1, \dots, n\}$,时隙长度定义为 $\Delta t = T/n$ 。此时,时隙长度应充分小,以保证在每个时隙内,网络拓扑固定不变^[14];然后,沿时间增长的方向,在相邻时隙同一卫星之间建立虚拟链路,表示数据的缓存过程。此时,LEO-SN可以表示为 $N=(N^T, E^T)$,其中 $N^T = \{N^1, \dots, N^n\}$ 为图 N 的节点集合; E^T 为图 N 的边集合。

3.2 星间链路加权时间扩展图

时间扩展图模型将LEO-SN动态拓扑转化为稳态图,完整地表征了任意时刻网络的连通情况,但缺少时间相关参数,不能精确计算给定数据量的传输时延。因此,本小节在时间扩展图的基础上,分别为时隙内和时隙间的边设计时延权重参数,构建WTEG模型,用于时延性能分析。时隙内和时隙间拓扑如图3所示。

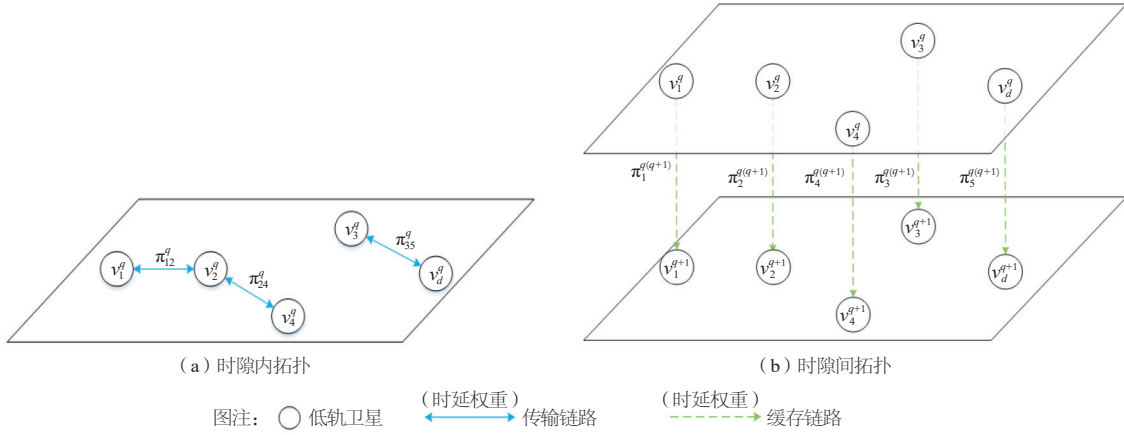


图3 时隙内和时隙间拓扑

时隙内边的权重: 任意时隙 $\forall q \in T$ 内, 边 $(v_p^q, v_o^q) \in E^T$ 的权重定义为单位数据量 D^* 由节点 v_p^q 传输到节点 v_o^q 的时延, 如式(7)所示。为降低模型复杂度, 本文忽略星间链路传播时延^[23]。则 q 时隙内所有边的权重可由邻接矩阵 G^q 表示, 如式(8)所示, 因此, q 时隙 LEO-SN 的拓扑可由图3(a)表示。

$$\pi_{po}^q = \begin{cases} \frac{D^*}{r^*}, \text{link}_{po}(\tau) = r^*, \tau = q \\ 0, \text{link}_{po}(\tau) = 0, \tau \neq q \end{cases} \quad (7)$$

$$G^q = \begin{bmatrix} 0 & \pi_{12}^q & \cdots & \pi_{1(d-1)}^q & \pi_{1d}^q \\ \pi_{21}^q & 0 & \cdots & \pi_{2(d-1)}^q & \pi_{2d}^q \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \pi_{(d-1)1}^q & \pi_{(d-1)2}^q & \cdots & 0 & \pi_{(d-1)d}^q \\ \pi_{d1}^q & \pi_{d2}^q & \cdots & \pi_{d(d-1)}^q & 0 \end{bmatrix} \quad (8)$$

时隙间边的权重: 由于 LEO-SN 拓扑不断变化, 不同时隙的链路连接情况不同, 数据在传输过程中, 可能会存在跨时隙的情况, 即在当前时隙内不能完成数据的传输。此时, 考虑到跨时隙拓扑的变化, 可能会存在链路中断的情况, 因此, 数据应缓存在当前中继卫星, 等待链路连通^[18], 本小节面向不同时隙的相同卫星, 提出了时隙间边的权重 $\pi_p^{q(q+1)}$, 定义为在数据传输到卫星节点 v_p^q 时当前时隙的剩余时间。只有明确数据到达卫星节点 v_p^q 时第 q 个时隙已消耗的时间 t_p^q , 才可由式(9)计算得到时隙间边的权重。特别地, 不同时隙的不同卫星之间不存在时隙间边的权重, 取值为 ∞ 。

$$\pi_p^{q(q+1)} = \Delta t - t_p^q \quad (9)$$

则相邻时隙 $(q, q+1) \in T$ 间所有边的权重可由邻接矩阵 $G^{q(q+1)}$ 如式(10)所示, 拓扑如图3(b)所示。

$$G^{q(q+1)} = \begin{bmatrix} \pi_1^{q(q+1)} & \cdots & \infty & \cdots & \infty \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \infty & \cdots & \pi_p^{q(q+1)} & \cdots & \infty \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \infty & \cdots & \infty & \cdots & \pi_d^{q(q+1)} \end{bmatrix} \quad (10)$$

由于卫星运行轨迹具有周期性, 时隙 n 至时隙1的拓扑结构 G^n 应满足式(10)。此时, 时间扩展图所有边的权重可通过组合矩阵 Graph 表示如下

$$\text{Graph} = \begin{bmatrix} G^1 & G^{12} & \infty & \cdots & \infty & \infty \\ \infty & G^2 & G^{23} & \cdots & \infty & \infty \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \infty & \infty & \infty & \cdots & G^{n-1} & G^{(n-1)n} \\ G^{n1} & \infty & \infty & \cdots & \infty & G^n \end{bmatrix} \quad (11)$$

综上所述, LEO-SN 可以表示为 WTEG, 如图4所示。

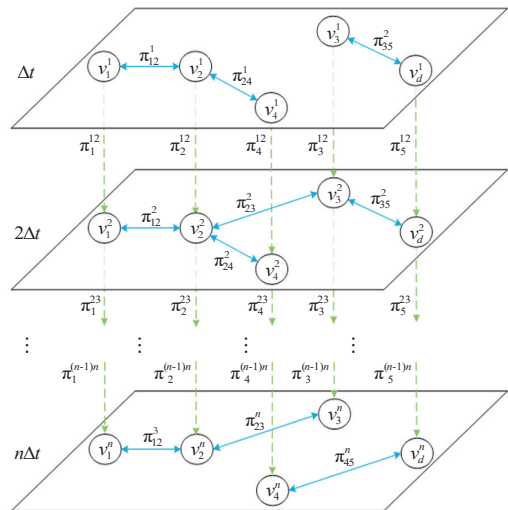


图4 WTEG模型

对任意节点对 $v_p^q, v_o^w \in N^T$, 将WTEG作为输入参数, 可通过最短路径算法 (如Dijkstra) 求得其最短时延 d_{po}^{qw} 和最短路由 path_{po}^{qw} , 设最短路由集合为 $P = \{\text{path}_{po}^{qw} | v_p^q, v_o^w \in N^T\}$ 。

4 业务图驱动的星间协作计算方案

4.1 星间业务调度规则

本小节采用图5 (a) 所示的业务DAG模型 $\Phi = (\Omega, \Gamma)^{[10]}$ 。其中 $\Omega = \{\omega_1, \dots, \omega_l\}$ 为节点集合, 表示构成 Φ 的子业务集群; Γ 为边集合, 表示子业务间的相互依赖关系。考虑到LEO-SN的特殊性, Φ 为单起点、单终点的DAG模型。此外, 子业务 $\omega_i \in \Omega$ 由元组 $\{D_i, \eta_i, C_i\}$ 表征, 其中 D_i 表示子业务 ω_i 的输入数据量, η_i 为数据缩放因子参数, 反映子业务 ω_i 执行过程中对输入数据量放大与缩小的过程, C_i 表示完成子业务 ω_i 所需的CPU周期数; 同时, 定义 $\varepsilon_i = (C_i/D_i)$ cycle/bit为计算复杂度系数^[20], 表征执行子业务 ω_i 耗费计算资源的程度, 其大小与所执行的任务有关; 定义 $\Theta_i(\omega_i) = \{\omega_j | (\omega_j, \omega_i) \in \Gamma\}$ 表示子业务 ω_i 的先驱节点集合, 则 D_i 可通过式(12)求得

$$D_i = \sum_{\omega_j \in \Theta_i(\omega_i)} D_j \eta_j \quad (12)$$

其中, D_j 和 η_j 属于表征服务功能 ω_j 的元组。此时, 业务 Φ 在LEO-SN中的调度规则可以转化为DAG至WTEG的映射规则。具体映射规则制定如下。

(1) Ω 至 N^T 的映射规则

定义B: $\Omega \rightarrow N^T$ 表示 Ω 至 N^T 的映射, 如式(13)所示。将业务起点映射至业务发起卫星; 将业务终点映射至业务处理结果接收卫星; 将中间节点映射至任意卫星。映射卫星计算子业务时可能会跨时隙, 此时数据会在该卫星上缓存, 经过虚拟缓存链路到达下一时隙。因此, 任意节点 $\omega_i \in \Omega$ 会被映射至 N^T 中同一卫星的 $\{\rho_i + 1 | \rho_i \in N\}$ 个副本节点, N 为自然数, ρ_i 表示 ω_i 计算所跨时隙的数目。

$$B(\omega_i) = \begin{cases} \{v_1^1, \dots, v_1^{1+\rho_i} | 1 + \rho_i \leq n\}, & \omega_i = \omega_1 \\ \{v_d^q, \dots, v_d^{q+\rho_i} | 1 \leq q, q + \rho_i \leq n\}, & \omega_i = \omega_l \\ \{v_p^q, \dots, v_p^{q+\rho_i} | 1 \leq q, q + \rho_i \leq n, 1 \leq p \leq d\}, & \text{其他} \end{cases} \quad (13)$$

(2) Γ 至 P 的映射规则

定义Z: $\Gamma \rightarrow P$ 表示 Γ 至 P 的映射, 将DAG的有向边映射到WTEG的无向边, 用来反映子业务之间的依赖关系, 如式(14)所示。Z将 $\forall (\omega_i, \omega_j) \in \Gamma$ 映射为图N节点集合 $B(\omega_i)$ 最后一个节点 $\text{end}(B(\omega_i))$ 到图N节点集合 $B(\omega_j)$ 第一个节点 $\text{start}(B(\omega_j))$ 之间的最短路由, 用 $\text{Path}_{\text{end}(B(\omega_i))\text{start}(B(\omega_j))}$ 表示

$$Z((\omega_i, \omega_j)) = \text{Path}_{\text{end}(B(\omega_i))\text{start}(B(\omega_j))} \quad (14)$$

基于上述定义, 本文给出一个DAG至WTEG的映射示

例, 如图5所示。此时, DAG被映射到WTEG中3个连续时隙上。值得注意的是, 业务发起卫星和结果接收卫星也可不为不同时隙的相同卫星, 则基于以上映射规则, 可得出业务发起卫星和结果接收卫星为同一颗卫星时的跨时隙映射方案。

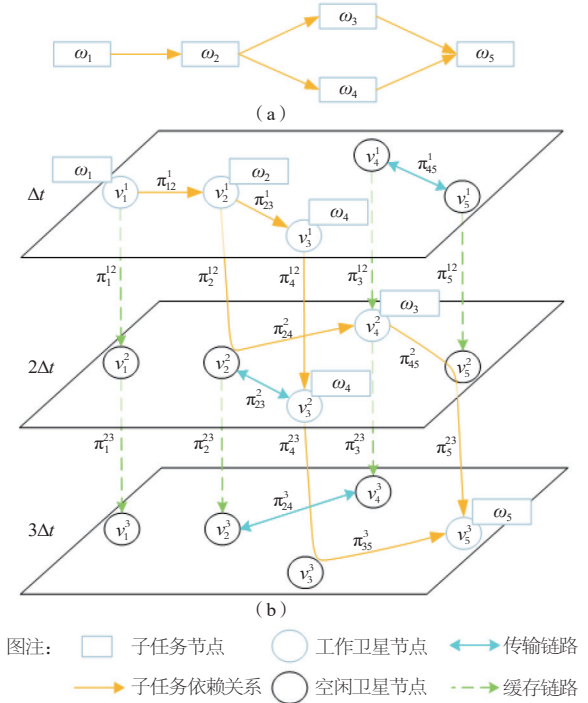


图5 DAG至WTEG的映射示例

4.2 时延优化问题

为了最小化业务处理时延, 基于4.1节的DAG至WTEG的调度规则, 本小节以业务处理时延为优化目标构建优化问题。为了便于描述又不失一般性, 本小节忽略业务请求由用户至发起卫星及业务处理结果由接收卫星至用户的过程带来的时延。

进行到子任务 ω_i 时的处理时延如式(15)所示

$$T(\omega_i) = T_{\text{comp}}(\omega_i) + T_{\text{accu}}(\omega_i) \quad (15)$$

式(15)中, $T_{\text{comp}}(\omega_i)$ 表示 ω_i 的计算时延, 如式(16)所示, $\rho_{B(\omega_i)}$ 为节点 $B(\omega_i)$ 的计算能力, 表示了卫星节点 $B(\omega_i)$ 的CPU每秒运行的周期数, 单位为cycle/s。

$$T_{\text{comp}}(\omega_i) = (D_i \eta_i \alpha_i) / \rho_{B(\omega_i)} \quad (16)$$

式(15)中, $T_{\text{accu}}(\omega_i)$ 表示进行到 ω_i 时的累积时延。考虑到 $\forall \omega_j \in \Phi_i(\omega_i)$ 的计算过程, ω_j 到 ω_i 的传输时延应为节点v传输数据量 $D_j \eta_j$ 至 $\text{start}(B(\omega_i))$ 的最短时延, 故子业务 ω_i 的累积时延 $T_{\text{accu}}(\omega_i)$ 如式(17)所示

$$T_{\text{accu}}(\omega_i) = \max_{\omega_j \in \Phi_i(\omega_i)} \left[T(\omega_j) + d_{\text{end}(B(\omega_j))\text{start}(B(\omega_i))} D_j \eta_j \right] \quad (17)$$

由式(16)和式(17)可知,子任务 ω_i 的处理时延可表示为式(18)

$$T(\omega_i) = \max_{\omega_j \in \Phi_i(\omega_i)} \left[T(\omega_j) + d_{\text{end}(B(\omega_j))\text{start}(B(\omega_i))} D_j \eta_j \right] + \frac{D_i \eta_i \alpha_i}{\rho_{B(\omega_i)}} \quad (18)$$

因此, Φ 的业务处理时延如式(19)所示

$$T(\Phi) = \max_{\omega_i \in \Phi} \left[T(\omega_i) + d_{\text{end}(B(\omega_i))\text{start}(B(\omega_i))} D_i \eta_i \right] + \frac{D_i \eta_i \alpha_i}{\rho_{B(\omega_i)}} \quad (19)$$

显然,对于相同的图 Φ 与图 N ,存在多个映射结果同时满足4.1节的映射规则,进而导致不同的业务处理时延。因此,需要寻找最优的映射结果以最小化业务处理时延。对于任意的图 Φ 与图 N ,每个映射结果都可以由一个 l 行 $n \times d$ 列的决策矩阵 X 唯一确定,决策矩阵 X 如式(20)所示

$$X = \begin{bmatrix} x(\omega_1, v_1^1) & \cdots & x(\omega_1, v_d^1) & \cdots & x(\omega_1, v_d^n) \\ x(\omega_2, v_1^1) & \cdots & x(\omega_2, v_d^1) & \cdots & x(\omega_2, v_d^n) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x(\omega_l, v_1^1) & \cdots & x(\omega_l, v_d^1) & \cdots & x(\omega_l, v_d^n) \end{bmatrix} \quad (20)$$

元素 $x(\omega_i, v_p^q) \in X$ 表示子任务 ω_i 是否被映射到节点 v_p^q 。

当 $x(\omega_i, v_p^q)=1$ 时,子任务 ω_i 被映射至节点 v_p^q ;反之,当 $x(\omega_i, v_p^q)=0$ 时,子任务 ω_i 不被映射至节点 v_p^q 。

$$x(\omega_i, v_p^q) \in \{0,1\}, \forall \omega_i \in \Omega, \forall v_p^q \in V^T \quad (21)$$

此外,由式(13)可知, ω_i 被映射到图 N 中 ρ_i+1 个节点,所以 $x(\omega_i, v_p^q) \in X$ 还应满足式(22)

$$\sum_{v_p^q \in V^T} x(\omega_i, v_p^q) = \rho_i + 1, \forall \omega_i \in \Omega \quad (22)$$

由式(19)~式(22)可得,业务处理时延可用式(23)表示,其中 $x(\omega_i, v_p^{q+\rho_i})$ 表示 ω_i 在图 N 中映射的最后一个节点, $x(\omega_i, v_d^n)$ 表示 ω_i 在图 N 中映射的第一个节点。

$$T(X) = \max_{\omega_i \in \Phi} \left[T(\omega_i) + \sum_{v_p^{q+\rho_i}, v_d^n \in V^T} d_{v_p^{q+\rho_i} v_d^n} D_i \eta_i x(\omega_i, v_p^{q+\rho_i}) x(\omega_i, v_d^n) \right] + \sum_{v_d^n \in V^T} \frac{D_i \eta_i \alpha_i}{\rho_{v_d^n}} x(\omega_i, v_d^n) \quad (23)$$

综上所述,LEO-SN的最佳星间协作计算方案建模如下

$$X = \arg \min(T(X))$$

$$\text{s.t.} \begin{cases} \text{Path}_{\text{end}(B(\omega_i))\text{start}(B(\omega_j))} \in P, & (\omega_i, \omega_j) \in \Gamma \\ \sum_{v_p^q \in V^T} x(\omega_i, v_p^q) = \rho_i + 1, & \forall \omega_i \in \Omega \\ \rho_i \in N, & \forall i \in \{1, \dots, l\} \\ x(\omega_i, v_p^q) \in \{0,1\}, & \forall \omega_i \in \Omega, v_p^q \in V^T \end{cases} \quad (24)$$

4.3 二值粒子群优化算法

二值粒子群优化(Binary Particle Swarm Optimization, BPSO)算法主要用于优化离散空间的约束问题,适用于式(24)提出的二值优化问题。本文假设粒子群规模为 U ,同时 I 表示群体最大迭代次数,则第 $i \leq I$ 次迭代过程中第 $u \leq U$ 个粒子的位置和速度分别为

$$X_u^i = (x_u^i(\omega_1, v_1^1), \dots, x_u^i(\omega_i, v_p^q), \dots, x_u^i(\omega_i, v_d^n)) \quad (25)$$

$$V_u^i = (v_u^i(\omega_1, v_1^1), \dots, v_u^i(\omega_i, v_p^q), \dots, v_u^i(\omega_i, v_d^n)) \quad (26)$$

式(25)中, $X_u^i \in K$, $K \triangleq \prod_{j=1}^{l \times n \times d} \{0,1\}$ 。式(26)中, $V_u^i \in O$, $O \triangleq \prod_{j=1}^{l \times n \times d} [-1,1]$ 。当 $i < I$ 时,粒子 u 速度更新为式(27)

$$V_u^{i+1} = \mu V_u^i + \beta_1 \gamma_1 (p_{u_{\text{best}}} - X_u^i) + \beta_2 \gamma_2 (g_{\text{best}} - X_u^i) \quad (27)$$

其中, μ 为惯性权重, γ_1, γ_2 为加速因子, β_1, β_2 为均匀分布于区间 $[0,1]$ 的随机数;随后粒子的位置更新为式(28)

$$X_u^{i+1} = X_u^i \otimes \lceil \max(V_u^{i+1}) \rceil \quad (28)$$

符号“ $\lceil \cdot \rceil$ ”表示数值向上取整运算,而“ $\otimes$ ”表示两个矩阵的元素之间的置换操作。最终,通过式(29)计算适应度值

$$f(X_u^{i+1}) = T(X_u^{i+1}) \quad (29)$$

综上所述,BPSO算法的具体步骤如下。

算法1 BPSO算法

输入: 输入一个粒子群 U , I , μ , γ_1 , γ_2 , β_1 , β_2

输出: $f(g_{\text{best}})$

1: for每个粒子 $u \leq U$ do

2: 初始化 X_u^0, V_u^0

3: 用式(29)评估粒子的适应度函数 u

4: 将当前位置设置为粒子的最佳位置 $p_{u_{\text{best}}}$

5: end for

6: 选择所有粒子中具有最佳适应性的粒子位置作为全局最佳位置 g_{best}

7: for generation 1 to I do

8: for每个粒子 $u \leq U$ do

9: 用式(27)更新粒子的速度 u

10: 用式(28)更新粒子的位置 u

11: 用式(29)评估新位置适应度值

12: if $f(X_u^{i+1}) < f(p_{u_{\text{best}}})$ then

13: $p_{u_{\text{best}}} = X_u^{i+1}$

14: end if

15: if $f(p_{u_{\text{best}}}) < f(g_{\text{best}})$ then

16: $g_{\text{best}} = p_{u_{\text{best}}}$

17: end if

18: end for

19: end for

20: return $f(g_{\text{best}})$

5 仿真实验及分析

为了验证本文提出的星间协作计算方案的优越性,本节首先对比星间协作计算与地面云计算的业务处理时延;接着,分析计算复杂度系数变化对业务处理时延的影响;然后,比较星间协作计算方案对不同数据缩放因子组合业务的承载能力;最后,比较BPSO算法和其他算法的时延性能。

仿真平台采用MATLAB,实验室计算机CPU为i5-7400,内存为24 GB。考虑到太空中即将部署数万颗LEO卫星,彼时LEO-SN拓扑切换将十分频繁,所以仿真设计时隙长度为100 ms,并且将LEO卫星的初始位置设置在拓扑切换的临界点。仿真采用的业务 Φ 如图6所示,且对于 $\forall \omega_i \in \Omega$, $\varepsilon_i = (1900/8)$ cycle/bit, $\eta_i = 1$ 。其余基本参数设置见表1^[20-22]。

表1 基本参数

参数	值	参数	值
G_r	37.5 dB	G_t	52.5 dB
P_t	5 kW	σ^2	5×10^{-15}
B_s	1 000 Mbit/s	ρ_s	[5,10] GHz
B_c	50 Mbit/s	ρ_c	100 GHz
r^*	1 000 Mbit/s	λ	0.05 m

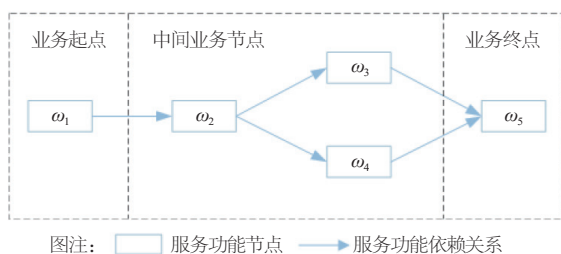


图6 业务 Φ

表1中, B_{sc} 为星地链路带宽, ρ_c 为地面云服务器计算能力, ρ_s 为卫星计算能力。此时卫星数量为12颗,卫星高度为750 km,星间距可根据式(2)计算得出,基于式(3)~式(11),LEO-SN前3个时隙的WTEG模型如图7所示。BPSO算法的基本参数为:种群规模 $U=100$,最大迭代次数 $I=50$,加速因子 $r_1=r_2=1$,惯性权重 $\mu=1.5$ 。此外,为不失一般性,任务量大小设置为[0,5] MB^[24]。

5.1 星间协作计算与地面云计算业务处理时延性能对比分析

本小节比较星间协作计算与地面云计算的业务处理时延,仿真结果如图8所示。

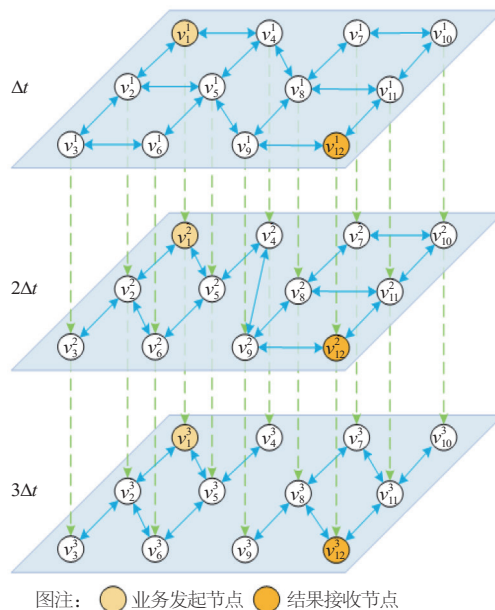


图7 前3个时隙的WTEG模型

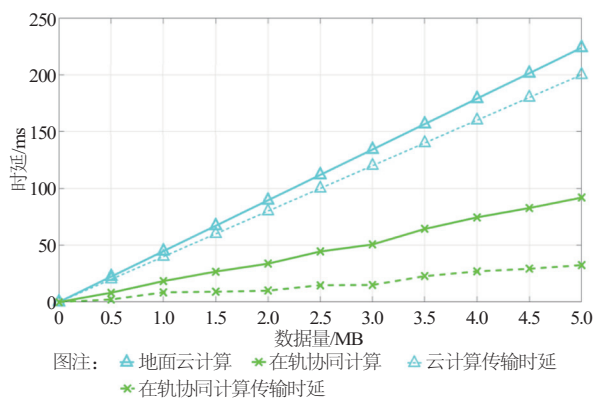


图8 星间协作计算与地面云计算的业务处理时延对比

仿真结果表明,星间协作计算的时延性能优于地面云计算,一方面,由于在地面云计算模式中,星地链路长距离数据传输会带来超高的传输时延,然而星间协作计算将业务调度到计算路径中多个卫星上,避免了星地长距离传输;另一方面,两者的差距呈现出越来越大的趋势,这是因为在数据量较小时,地面云服务器的计算能力可以弥补传输时延造成的损失,但是随着数据量不断增加,传输时延的影响越发明显。具体而言,当数据量较小,如1 MB时,星间协作计算与地面云计算的业务处理时延分别为18.38 ms和44.75 ms,此时两者的时延差仅为26.37 ms。随着数据量由1 MB增加至5 MB,星间协作计算的业务处理时延平均增速为18.37 ms/MB,计算时延增速稍高于传输时延增速,两者分别为11.87 ms/MB和6.5 ms/MB;相反地,地面云计算的业务处理时延增速一

直居高不下, 稳定在44.75 ms/MB, 其中传输时延增速高达40 ms/MB, 占处理总时延的89.4%, 是处理时延增速居高不下的主要原因。当数据量增至5 MB时, 两者的处理时延分别为91.87 ms和223.75 ms, 两者的时延差已增至131.88 ms。此时, 相比地面云计算, 星间协作计算将时延性能提高了58.9 %。因此, 星间协作计算方法更易满足业务的低时延需求。

5.2 业务处理时延与计算复杂度系数变化关系

本小节分析计算复杂度对业务处理时延的影响。以业务 Φ 为例, 当其计算复杂度参数从 $0.2\epsilon_i$ 增加到 $1.8\epsilon_i$ 时, 业务处理时延的变化趋势如图9所示。

由图9可知, 若数据量一定, 随着计算复杂度升高, 业务处理时延逐渐增加, 而且, 业务处理时延的增加在不同数据量约束下呈现两种趋势。当数据量为1 MB时, 随着计算复杂度由 $0.2\epsilon_i$ 升高到 $1.8\epsilon_i$, 业务总时延由6.4 ms平缓增加至27.9 ms。这是因为随着计算复杂度升高, 业务的计算时延相应升高引起业务处理时延相应增加; 然而, 当数据量为5 MB时, 随着计算复杂度升高, 业务总时延由50.9 ms跳跃式增加至152.3 ms。特别是, 业务处理时延在复杂度为 $1.4\epsilon_i$ 时由98.8 ms跳跃式增加至125.5 ms。这是因为当计算复杂度增加到 $1.4\epsilon_i$ 时, 由于计算时延增长, Φ 不能在原本的时隙完成, 数据需要通过虚拟缓存链路到达下一时隙, 此缓存过程导致传输时延显著增加, 最终, 业务处理时延出现跳跃式增长。

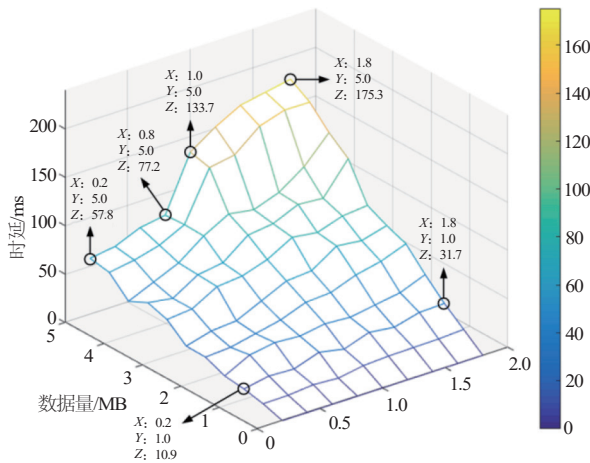


图9 计算复杂度参数对业务处理时延的影响

5.3 业务处理时延与数据缩放因子变化关系

本小节比较星间协作计算对不同数据缩放因子组合业务的承载能力。显然, 图5中DAG的节点集合可以被划分为前排节点 (即 ω_1, ω_2) 和后排节点 (即 $\omega_3, \omega_4, \omega_5$)。

按照不同的数据缩放因子组合, 业务可以被分为前缩后放型和前放后缩型。仿真设计了4种业务 Φ_1 、 Φ_2 、 Φ_3 、 Φ_4 与 Φ 做对比, Φ_1 、 Φ_2 、 Φ_3 、 Φ_4 的数据缩放因子组合分别设置为 $\{0.5, 1, 1, 1, 2\}$ 、 $\{2, 1, 1, 1, 0.5\}$ 、 $\{1, 1, 0.5, 2, 1\}$ 、 $\{1, 1, 2, 0.5, 1\}$, 其中 Φ_3 、 Φ_4 是为了讨论在并行业务分支上数据一缩一放的情况对业务处理时延的影响, 仿真结果如图10所示。

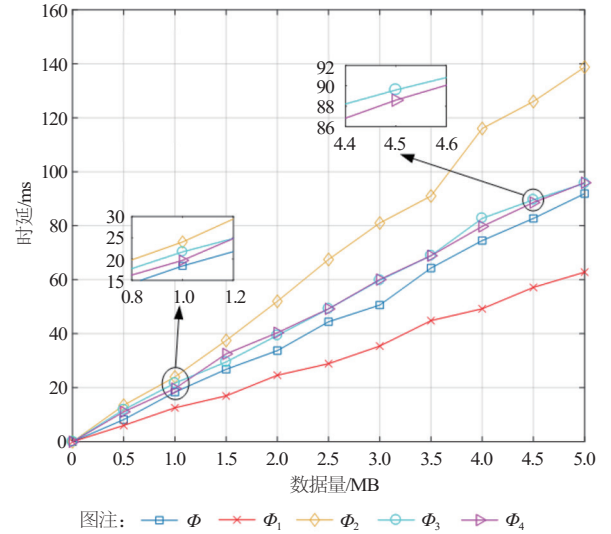


图10 不同数据缩放因子组合业务时延对比

由图10可知, 当数据量为5 MB时, Φ 、 Φ_1 、 Φ_2 、 Φ_3 与 Φ_4 的业务处理时延分别为91.87 ms、62.81 ms、138.75 ms、95.68 ms和95.93 ms。显然, Φ_1 的业务处理时延最低, 这是因为其在业务起点将数据量压缩为输入数据量的一半, 使得后续过程所需处理的数据量减半。同理, 由于业务起点将数据量放大为输入数据量的两倍, 导致后续过程所需处理的数据量翻倍, 所以 Φ_2 的业务处理时延最高。此外, Φ_3 与 Φ_4 的业务处理时延相当且略高于 Φ 的总时延。由式(17)可知, ω_5 的累积时延为 ω_3 与 ω_4 引起的累积时延的最大值, 故当输入数据量和计算复杂度相等时, ω_5 的累积时延由分支中最大的数据缩放因子决定。综上所述, 星间协作计算方案在处理前缩后放型业务时更能发挥其性能优势, 而且数据缩小因子更宜分布在DAG主干节点。

5.4 多种算法实验性能比较

本小节比较BPSO算法同加权轮转法 (WRR)、随机动态算法 (Pick-KX)、贪婪负载均衡算法 (Greedy-LB) 的时延性能, 仿真结果如图11所示。

由图11可知: 当数据量 $D < 1.5$ MB时, 4种算法时延差别不明显; 当任务量 $D > 2$ MB时, 随着任务量的增加,

BPSO算法的时延明显低于其他3种算法。例如：当任务量为5 MB时，BPSO、Greedy-LB、WRR、Pick-KX的时延分别为91.875 ms、141.25 ms、155.375 ms、181.5 ms，BPSO算法相比于其他3种算法分别降低了34.96%、40.87%、49.38%。

这是因为Pick-KX算法随机选取LEO卫星进行映射，并未考虑其计算能力；WRR算法根据卫星的计算能力大小关系进行映射，Greedy-LB算法通过选取当前负载最小的卫星节点进行映射，但WRR和Greedy-LB均未考虑网络边的传输能力，仍不是最优；BPSO算法综合考虑节点计算能力和网络边的传输能力，可有效降低任务处理时延。

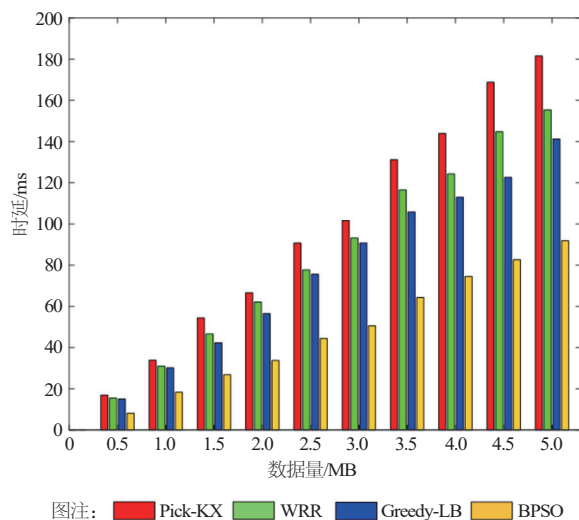


图11 多种算法时延性能比较

6 结束语

BPSO算法被用于求解本文中的NP难问题，仿真结果证明了该方案的有效性。但本文仅考虑了单用户单业务场景，没有考虑到多个用户或者多个业务的情况。此外，本研究选用的业务DAG模型较为单一，没有展开研究不同的DAG模型对该方法的影响，目前比较流行的DAG模型通常比较复杂。因此，多业务多用户场景及复杂的DAG模型内在结构对该方法时延性能的影响是未来的研究重点。

参考文献：

- [1] BACCO M, DAVOLI F, GLAMBERE G, et al. Networking challenges for non-terrestrial networks exploitation in 5G[C]//2019 IEEE 2nd 5G World Forum (5GWF). Piscataway: IEEE Press, 2019: 623-628.
- [2] 王艳峰, 谷林海, 刘鸿鹏. 低轨卫星移动通信现状与未来发展[J]. 通信技术, 2020, 53(10): 2447-2453.
WANG Y F, GU L H, LIU H P. Statusquo and future development of LEO satellite mobile communication[J]. Communications Technology, 2020, 53(10): 2447-2453.
- [3] 靳登匀. 面向OneWeb卫星网络的SDN控制器设计与实现[D]. 哈尔滨: 哈尔滨工业大学, 2019.
JIN D Y. Design and implementation of SDN controller for OneWeb satellite network[D]. Harbin: Harbin Institute of Technology, 2019.
- [4] HOFMANN A, GLEIN R, FRANK L, et al. Reconfigurable on-board processing for flexible satellite communication systems using FPGAs[C]//2017 Topical Workshop on Internet of Space (TWIOS). Piscataway: IEEE Press, 2017: 1-4.
- [5] JACOBS A, CONGER C, GEORGE A D. Multiparadigm space processing for hyperspectral imaging[C]//2008 IEEE Aerospace Conference. Piscataway: IEEE Press, 2008: 1-11.
- [6] WANG F, JIANG D, QI S, et al. Dynamic computing resource adjustment in edge computing satellite networks[C]//International Conference on Simulation Tools and Techniques. Springer, 2019: 135-145.
- [7] BRAVHAR K, MARTINS V, SANTOS L, et al. BRAVE NG-MEDIUM FPGA reconfiguration through Space Wire: example use case and performance analysis[C]//2018 NASA/ESA Conference on Adaptive Hardware and Systems (AHS). Piscataway: IEEE Press, 2018: 135-141.
- [8] 何秀丽, 任智源, 史晨华, 等. 面向医疗大数据的云雾网络及其分布式计算方案[J]. 西安交通大学学报, 2016, 50(10): 71-77.
HE X L, REN Z Y, SHI C H, et al. A cloud and fog network architecture for medical big data and its distributed computing scheme[J]. Journal of Xi'an Jiaotong University, 2016, 50(10): 71-77.
- [9] 叶波. 基于负载均衡度的云计算任务调度算法[J]. 东北电力大学学报, 2019, 39(1): 88-95.
YE B. Task scheduling algorithm for cloud computing based on load balance degree[J]. Journal of Northeast Electric Power University, 2019, 39(1): 88-95.
- [10] SHAH V, DEY B K, MANJUNATH D. Network flows for function computation[J]. IEEE Journal on Selected Areas in Communications, 2013, 31(4): 714-730.
- [11] ROST M, SCHMID S. Virtual network embedding approximations: Leveraging randomized rounding[J]. IEEE/ACM Transactions on Networking, 2019, 27(5): 2071-2084.
- [12] MICHAEL M, LLORCA J, TULINO A. Approximation algorithms for the optimal distribution of real-time stream-

- processing services[C]//2019 IEEE International Conference on Communications (ICC). Piscataway: IEEE Press, 2019: 1-7.
- [13] KAMINSKI N, MACALUSO I, PASCALE E D, et al. A neural-network-based realization of in-network computation for the Internet of Things[C]//2017 IEEE International Conference on Communications (ICC). Piscataway: IEEE Press, 2017: 1-6.
- [14] WANG P, ZHANG X, ZHANG S, et al. Time-expanded graph-based resource allocation over the satellite networks[J]. IEEE Wireless Communications Letters, 2019, 8(2): 360-363.
- [15] SHENG M, WANG Y, LI J, et al. Toward a flexible and reconfigurable broadband satellite network: Resource management architecture and strategies[J]. IEEE Wireless Communications, 2017, 24(4): 2-8.
- [16] CAO S, ZHAO Y, WEI J, et al. Space-Based Cloud-Fog Computing Architecture and Its Applications[C]//2019 IEEE World Congress on Services (SERVICES). Piscataway: IEEE Press, 2019: 166-171.
- [17] XUE R, YU H, CHENG Q. Adaptive coded modulation based on continuous phase modulation for inter-satellite links of global navigation satellite systems[J]. IEEE Access, 2018, 6: 20652-20662.
- [18] FALL K, FARRELL S. DTN: An architectural retrospective[J]. IEEE Journal on Selected Areas in Communications, 2008, 26(5): 828-836.
- [19] WANG Y, SHENG M, WANG X, et al. Mobile-edge computing: Partial computation offloading using dynamic voltage scaling[J]. IEEE Transactions on Communications, 2016, 64(10): 4268-4282.
- [20] XU S, WANG X, HUANG M. Capacity analysis method for MLSN based on improved DGA[C]//2015 11th International Conference on Natural Computation (ICNC). Piscataway: IEEE Press, 2015: 215-220.
- [21] CAO S Z, WEI J Y, HAN H, et al. Space edge cloud enabling network slicing for 5G satellite network[C]//2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC). Piscataway: IEEE Press, 2019: 787-792.
- [22] HOU X, REN Z, WANG J, et al. Distributed fog computing for latency and reliability guaranteed swarm of drones[J]. IEEE Access, 2020(8): 7117-7130.
- [23] HAN J, WANG H, WU S, et al. Task scheduling of high dynamic edge cluster in satellite edge computing[C]//2020 IEEE World Congress on Services (SERVICES). Piscataway: IEEE Press, 2020.
- [24] WANG Y, ZHANG J, ZHANG X, et al. A computation offloading strategy in satellite terrestrial networks with double edge computing[C]//2018 IEEE International Conference on Communication Systems (ICCS). Piscataway: IEEE Press, 2018.

[作者简介]



国晓博（1985—），中国电子科技集团第五十四研究所高级工程师，主要研究方向为天地一体化信息网络、软件定义网络。



任智源（1983—），西安电子科技大学副教授，硕士生导师，主要研究方向为面向卫星网络、无人机网络等高动态场景的网络计算，业务驱动的可编程网络设计与开发平台等。



程文驰（1986—），西安电子科技大学教授，博士生导师，主要研究方向为后5G/6G通信网络关键技术、无线网络时延管控、全双工、卫星互联网等。



纪哲（1989—），北京邮电大学信息与通信工程学院讲师，主要研究方向为空天信息网络、机器学习等。