

DOI: 10.7652/xjtuxb202002015

面向医疗大数据任务低时延需求的路径计算方案

张夏童¹, 任智源¹, 胡锦涛¹, 张怡², 张海林¹

(1. 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, 710071, 西安;
2. 中国空间技术研究院西安分院空间微波技术重点实验室, 710100, 西安)

摘要: 针对云计算应用于医疗大数据处理时存在的通信高负荷及任务处理高时延的问题,提出一种面向低时延任务需求的路径计算方案。该方案首先将医疗大数据任务构建为由多个具有输入输出关系的子任务所组成的有向无环图;然后,设计了一种云雾网络架构,利用医院中的交换机、路由器等网络边缘设备组成雾计算层,在端到端的定向数据传输过程中利用雾节点的计算能力逐步完成医疗大数据任务计算。为将医疗大数据任务部署至医院网络,提出一种基于离散二值粒子群优化(BPSO)算法的任务映射策略,将有向无环图形式的大数据任务映射至医院雾网络拓扑图,为任务数据寻找合适的计算路径,并最小化医疗大数据任务处理时延。仿真结果表明,当数据量取 5~10 Mb 时,应用路径计算方案的任务处理时延相比云计算可降低 50% 以上。

关键词: 医疗大数据;云雾网络;映射;路径计算

中图分类号: TM912 **文献标志码:** A **文章编号:** 0253-987X(2020)02-0119-08

A Path Computing Scheme for Low-Latency Requirement of Medical Big Data Task

ZHANG Xiatong¹, REN Zhiyuan¹, HU Jintao¹, ZHANG Yi², ZHANG Hailin¹

(1. State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China;
2. National Key Laboratory of Science and Technology on Space Microwave (Xi'an), CAST, Xi'an 710100, China)

Abstract: A path computing scheme for low-latency task is proposed to solve the problem that there exist high communication load and high task processing latency when cloud computing is applied to medical big data processing. The scheme firstly constructs the medical big data task into a directed acyclic graph that is composed of multiple subtasks with explicit input and output relationships. Then, a cloud and fog network architecture is designed, in which network edge devices such as switches and routers in hospitals form a fog computing layer. The computing capacity of fog nodes is used to gradually complete the big data task in the process of end-to-end directional data transmission. A task mapping strategy based on the discrete binary particle swarm optimization (BPSO) algorithm is proposed to deploy the medical big data task to a hospital network. The big data task in the form of a directed acyclic graph is mapped into a topology graph of the hospital fog network to find an appropriate computing path for the task and to minimize the latency. Simulation results and a comparison with the cloud computing show that the proposed path computing scheme reduces the latency by more than 50% when the amount of

data is 5-10 Mb.

Keywords: medical big data; cloud and fog network; mapping; path computing

近年来,医疗大数据凭借其大容量、高价值、快速增长及多样化的特点,在流行病预测、健康管理及精准诊疗等领域呈现了极大的价值^[1-2]。过去十年间,谷歌与斯坦福大学先后在流感趋势预测、艾滋病检测方面建立了医疗大数据分析平台^[3-4]。现阶段,各国政府均已认识到医疗大数据作为国家基础性战略资源的重要性,争相抢占医疗大数据研究的前沿阵地^[5],但规模迅速增长的医疗大数据对硬件基础设施的计算和存储能力提出了更高的要求^[6]。目前,云计算凭借其强大的计算和存储能力已经发展成为主流的医疗大数据分析处理支撑平台^[7]。然而,海量的数据传输会给现有基础设施带来巨大的通信压力,大数据处理系统的鲁棒性较差^[8],同时,云端与医院本地终端之间的远距离数据交互过程也会产生较高的数据传输时延,诊断效率无法保障。

针对云计算模式存在的上述问题,研究者们提出了路径计算技术,利用数据传输路径上通信/转发节点的计算能力,在数据传输的过程中完成大数据任务处理。关于路径计算的早期研究工作主要是在带噪广播网络中计算特定数学函数所需数据量的渐近性分析^[9-11]。随后,为了优化通信网的路径计算速率,研究者们将计算任务图与网络拓扑图之间的映射关系视为新的研究重点。Virag 等构建了一个线性规划问题研究网络中的数据流路由方案来最大化通信网的路径计算速率^[12]。针对文献^[12]中任务只能表示为计算树的不足,Samta 等研究了当任务的表示形式为若干子任务组成的有向无环图时的路径计算速率优化问题^[13],但其并未考虑通信节点的计算能力限制。目前,在路径计算技术的应用方面,Dejan 等在原有 3GPP 的机器类型通信(MTC)架构基础上增加了网络函数计算(NFC)服务,提出基于软件定义网络和网络功能虚拟化技术的 MTC-NFC 架构^[14]。此外,Nichoas 等提出了一个基于物联网的神经网络计算框架,将神经网络的神经元运算映射至物联网节点上完成^[15]。然而,在医疗大数据处理与路径计算理论的融合方面,暂未见到公开发表的论文与技术成果。

为了克服云计算模式存在的鲁棒性差及时延高的弊病,本文将路径计算技术引入医疗大数据领域,提出一种适应低时延任务需求的路径计算方案。首先,设计一种云雾网络架构,充分利用医院中的交换

机、路由器等网络边缘设备的计算能力,在进行端到端数据传输的同时,分步骤完成医疗大数据的任务计算;其次,制定任务映射策略,将有向无环图(DAG)形式的大数据任务映射至医院的无向图(UG)网络中,并基于离散二值粒子群优化算法(BPSO)求解最优映射关系。仿真结果表明,本文路径计算方案能够有效降低 DAG 形式的医疗大数据任务处理时延,保障医生工作效率与病人就诊体验。

1 云雾网络架构

为了能够利用医院现有基础设施支撑路径计算方案的实现,本文面向医疗大数据场景,借鉴文献^[16],首先提出一种云雾网络架构,如图 1 所示。该网络架构自上而下分为 3 层,分别为云计算层、雾计算层、终端用户层。云计算层由高性能的服务器集群组成;雾计算层由医院中的交换机、路由器等网络边缘设备组成,一方面为医疗设备和用户提供网络接入通道,并负责整个雾网络中数据转发,另一方面,通过主动缓存的方式从云计算层拉取计算服务至本地,进行任务计算,终端用户层由诸多医疗检测设备与目标用户终端组成。基于此架构,医疗大数据无需传输至云端进行处理,而是通过医院的雾网络直接发送给目标用户,利用数据传输路径上预先缓存好的计算服务逐步完成大数据任务计算,目标用户获取到的数据即为病理分析结果。

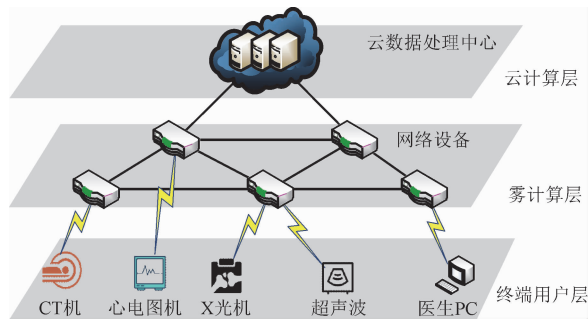


图1 云雾网络架构

2 任务映射策略

为了能够将路径计算方案应用于医疗大数据处理过程,需将医疗大数据任务部署至医院雾网络。现有的医疗大数据任务处理过程往往无法用一个独立的单任务表示,通常可以将其分割为多个离散的

子任务,且这些子任务之间有明确的输入输出逻辑关系,因而可以采用 DAG 进行表示。因此,路径计算方案要解决的首要问题是如何将 DAG 形式的大数据任务映射至 UG 形式的医院雾网络中,映射示例如图 2 所示。

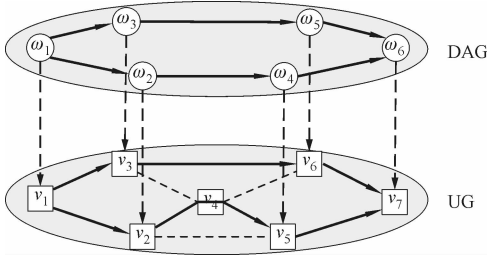


图2 DAG至UG映射示例

2.1 DAG至UG的映射规则

本小节用离散子任务组成的有向无环图 $G = (\Omega, \Gamma)$ 表示医疗大数据任务模型,其中 $\Omega = \{\omega_1, \omega_2, \dots, \omega_s, \omega_{s+1}, \dots, \omega_{l-1}, \omega_l \mid s \geq 1, l \geq s+1\}$ 为图 G 的节点集合,即子任务节点集合 $\omega_1, \omega_2, \dots, \omega_s$ 为 s 个任务起点, $\omega_{s+1}, \dots, \omega_{l-1}$ 为中间子任务处理节点, ω_l 为任务终点; Γ 为图 G 的边集合。此外,定义 $\Phi_i(\omega_i) = \{\omega_j \mid (\omega_j, \omega_i) \in \Gamma\}$ 为子任务节点 ω_i 的前向节点集合。由于 DAG 形式的医疗大数据任务中可能存在多个并行计算的任务处理过程,所以本小节假设图 G 由多个任务分支组成,每个任务分支均由若干出度为 1 的子任务节点串行组成,代表多个同时进行的任务处理过程,如图 3 所示。

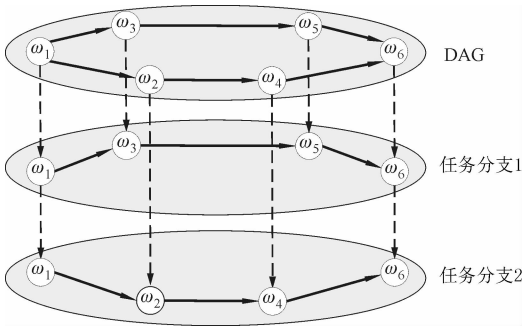


图3 DAG中的任务分支结构

同时,本小节用无向连通图 $N = (V, E)$ 表示医院网络拓扑图,其中, $V = \{v_1, v_2, \dots, v_s, v_{s+1}, \dots, v_{t-1}, v_t \mid s \geq 1, t \geq s+1\}$ 为图 N 的节点集合,即医院雾网络节点, $v_1, v_2, \dots, v_s$ 为实际的数据源节点,即医疗检测设备的直连网络节点, v_t 为目标用户的直连网络节点, E 为图 N 的边集合。定义集合 $C = \{v_1, v_2, \dots, v_s\} \cup \{v_t\}$ 。

对于图 N ,假设边集合 E 的每条无向边均支持双向数据传输,且节点集合 V 中任意两节点间至少存在一条连接通路。此外,定义 d_{v_i, v_j} 为从节点 v_i 传输单位数据量至节点 v_j 的最短时延,定义能够以最短时延 d_{v_i, v_j} 从节点 v_i 传输单位数据量至节点 v_j 的通路为节点 v_i 至节点 v_j 最短通路,用 $\text{path}(v_i, v_j)$ 表示;同时,定义 $P = \{\text{path}(v_i, v_j) \mid v_i, v_j \in V\}$ 为图 N 中所有节点对的最短通路集合;对于任意节点对 $v_i, v_j \in V$,将图 N 的网络边数据传输速率与节点连接关系作为参数,通过 Dijkstra 算法求得最短时延 $d_{v_i, v_j}, d_{v_j, v_i}$ 与最短通路 $\text{path}(v_i, v_j), \text{path}(v_j, v_i)$ 。

本文所提路径计算方案需将图 G 中的每个子任务节点映射为图 N 中的单个网络节点来执行其计算过程,借鉴文献[17],对于给定图 G 及图 N 的映射规则如下。

定义1 子任务节点集合 Ω 至网络节点集合 V 的映射为 $B: \Omega \rightarrow V$,且 B 需满足式(1)条件

$$B(\omega_i) = \begin{cases} v_i, & i \in \{1, 2, \dots, s\} \\ v_t, & i = l \\ v_j, & v_j \in V \setminus C, \text{其他} \end{cases} \quad (1)$$

B 将集合 Ω 的任务起点 $\omega_1, \omega_2, \dots, \omega_s$ 依次映射为集合 V 的数据源节点 $v_1, v_2, \dots, v_s$;将集合 Ω 的任务终点 ω_l 映射为集合 V 的节点 v_t ;将集合 Ω 的中间子任务处理节点 $\omega_{s+1}, \dots, \omega_{l-1}$ 映射为集合 V 的子集 $\{v_{s+1}, \dots, v_{t-1}\}$ 中的任意节点。

定义2 有向边集合 Γ 至网络节点对最短通路集合 P 的映射为 $Z: \Gamma \rightarrow P$,且 Z 需满足式(2)条件

$$Z((\omega_i, \omega_j)) = \text{path}(B(\omega_i), B(\omega_j)) \quad (2)$$

Z 将集合 Γ 中的有向边 (ω_i, ω_j) 映射为图 N 中节点 $B(\omega_i)$ 至节点 $B(\omega_j)$ 的最短通路 $\text{path}(B(\omega_i), B(\omega_j))$ 。

2.2 DAG至UG的映射关系优化模型

对于同一对图 G 与图 N ,存在多种映射关系同时满足上述映射规则,当图 G 依据不同的映射关系映射至图 N 后会在图 N 中形成不同的任务计算路径,进而导致不同的任务处理时延。为了进一步降低医疗大数据任务处理时延,本文基于前述的 DAG 至 UG 的映射规则,构建出一个时延最小的优化问题,达到降低任务处理时延的目的。

首先,定义子任务 ω_i 在某次映射关系中的处理时延如下

$$T(\omega_i) = T_{\text{accu}}(\omega_i) + T_{\text{comp}}(\omega_i) \quad (3)$$

$$T_{\text{accu}}(\omega_i) = \max_{\omega_j \in \Phi_i(\omega_i)} [T(\omega_j) + d_{B(\omega_j)B(\omega_i)} D_{\omega_j \omega_i}] \quad (4)$$

$$T_{\text{comp}}(\omega_i) = \frac{D_{\omega_i} \alpha}{p_{B(\omega_i)}} \quad (5)$$

式中: $T_{\text{accu}}(\omega_i)$ 为总任务进行到子任务 ω_i 时的累积时延; $D_{\omega_j \omega_i}$ 为此次映射中网络通路 $\text{path}(B(\omega_i), B(\omega_j))$ 所需传输的数据量; $T_{\text{comp}}(\omega_i)$ 为网络节点 $B(\omega_i)$ 完成子任务 ω_i 所耗费的计算时延; D_{ω_i} 为此次映射中 $B(\omega_i)$ 所需计算的数据量; $p_{B(\omega_i)}$ 为网络节点 $B(\omega_i)$ 的计算能力; α 为任务计算复杂度系数。式(5)借鉴了文献[18]。另外, 将任务起点 $\omega_1, \omega_2, \dots, \omega_s$ 的处理时延限制为 0。

为了方便后续优化问题的建立, 本文假设每个任务分支具有不同的数据量权重因子 $\sigma_u (u=1, 2, \dots, K)$, σ_u 用来表示第 u 个分支所需传输、计算的数据量在任务总数据量中的占比, K 为图 G 中的任务分支数, 且所有数据量权重因子加和为 1。

由于大数据任务是在数据传输的过程中被逐步计算完成, 因此为准确模拟 DAG 形式的任务计算过程, 本文为所有子任务定义统一的数据量衰减系数 $\eta (0 < \eta < 1)$, 以表示子任务计算过程对于任务数据量的压缩程度。

由式(3)~式(5)可得子任务 ω_i 在某次映射中的处理时延

$$T(\omega_i) = \max_{\omega_j \in \Phi_i(\omega_i)} [T(\omega_j) + d_{B(\omega_j)B(\omega_i)} D_{\omega_j \omega_i}] + \frac{D_{\omega_i} \alpha}{p_{B(\omega_i)}} \quad (6)$$

图 G 的任务处理时延即为任务终点 ω_l 的处理时延

$$T(G) = T(\omega_l) \quad (7)$$

定义图 G 中的节点 ω_p 与图 N 中的节点 v_q 的映射关系如下

$$x_{\omega_p v_q} \in \{0, 1\}, \quad \forall \omega_p \in \Omega, \forall v_q \in V \quad (8)$$

当 $x_{\omega_p v_q} = 1$ 时, 节点 ω_p 被映射至节点 v_q ; 反之, 当 $x_{\omega_p v_q} = 0$ 时, 节点 ω_p 不会被映射至节点 v_q 。

考虑到式(8)中的节点映射关系, 式(6)所表示的子任务处理时延也可用下式表示

$$T(\omega_i) = \max_{\omega_j \in \Phi_i(\omega_i)} \left[T(\omega_j) + \sum_{v_p, v_q \in V} d_{v_p v_q} x_{\omega_j v_q} x_{\omega_i v_p} D_{\omega_j \omega_i} \right] + \sum_{v_p \in V} \frac{D_{\omega_i} \alpha}{p_{v_p}} x_{\omega_i v_p} \quad (9)$$

所有子任务节点与网络节点的映射关系可以构建成一个长度为 $l \times t$ 的一维向量 $\mathbf{X} = [x_{\omega_1 v_1}, \dots, x_{\omega_l v_j}, \dots, x_{\omega_l v_t}]$, $T(G)$ 也可以表示为 $\mathbf{X}$ 的函数

$$T(G) = F(\mathbf{X}) \quad (10)$$

$$\mathbf{X} = \arg \min (F(\mathbf{X}))$$

$$\text{s. t.} \begin{cases} x_{\omega_p v_q} \in \{0, 1\}, \forall \omega_p \in \Omega, \forall v_q \in V \\ \sum_{v_q \in V} x_{\omega_p v_q} = 1, \forall \omega_p \in \Omega \\ x_{\omega_p v_q} = 1, p = q, \forall p \in [1, s] \\ x_{\omega_p v_q} = 1, \omega_p = \omega_l, v_q = v_l \\ x_{\omega_p v_q} = 0, q \in [1, s], \forall p \in [s+1, l-1] \\ x_{\omega_p v_q} = 0, v_q = v_l, \forall p \in [s+1, l-1] \\ T(\omega_p) = 0, \forall p \in [1, s] \end{cases} \quad (11)$$

综上所述, 基于本文所提映射规则, 图 G 至图 N 的最优映射关系的求解可建模为式(11)表示的二值优化问题。

2.3 BPSO 算法

BPSO 算法与传统的粒子群优化 (PSO) 算法不同, 优化对象不是连续实值问题, 而是离散空间的约束问题。算法将粒子位置的每一维限制为 0 或 1, 适用于式(11)所述的二值优化问题, 故本文选用离散二值粒子群算法求解上述数学模型。

在求解式(11)所述优化问题时, 粒子群 $\{\mathbf{X}^n(i)\}_{i=1}^M$ 在搜索空间 $\mathbf{I}$ 内移动以寻找最佳位置, 其中 M 为粒子群规模, $n \in \{1, 2, \dots, N_{\max}\}$ 为迭代次数, $N_{\max}$ 为最大迭代次数。在第 n 次迭代过程中, 第 i 个粒子的位置和速度分别表示为

$$\mathbf{X}^n(i) = (x_{\omega_1 v_1}^n(i), \dots, x_{\omega_l v_j}^n(i), \dots, x_{\omega_l v_t}^n(i)) \quad (12)$$

$$\mathbf{V}^n(i) = (v_{\omega_1 v_1}^n(i), \dots, v_{\omega_l v_j}^n(i), \dots, v_{\omega_l v_t}^n(i)) \quad (13)$$

式中: $\mathbf{X}^n(i) \in \mathbf{I}, \mathbf{I} \triangleq \prod_{j=1}^{l \times t} \{0, 1\}; \mathbf{V}^n(i) \in \mathbf{O}, \mathbf{O} \triangleq \prod_{j=1}^{l \times t} [-1, 1]$ 。

在第 n 次迭代中, 粒子 i 跟踪局部最优和全局最优位置并更新其速度和位置的公式分别如下

$$\mathbf{V}^{n+1}(i) = \rho \mathbf{V}^n(i) + \beta_1 \gamma_1 (\mathbf{p}_{\text{best}}^n(i) - \mathbf{X}^n(i)) + \beta_2 \gamma_2 (\mathbf{g}_{\text{best}}^n(i) - \mathbf{X}^n(i)) \quad (14)$$

$$\mathbf{X}^{n+1}(i) = \mathbf{X}^n(i) + \mathbf{V}^{n+1}(i) \quad (15)$$

式中: $\mathbf{p}_{\text{best}}^n(i)$ 和 $\mathbf{g}_{\text{best}}^n(i)$ 分别为粒子的局部最优和全局最优位置; ρ 为惯性权重; γ_1, γ_2 为加速因子; β_1 和 β_2 分别为均匀分布于区间 $[0, 1]$ 的随机数。本文定义适应度函数

$$f(\mathbf{X}) = T(G) = F(\mathbf{X}) \quad (16)$$

综上所述, BPSO 算法求解上述优化问题的算法步骤如下:

```

(1) for  $i=1:M$  do
(2) 随机初始化粒子速度和位置  $V(i), X(i)$ 
(3) 通过式(16)计算适应度值  $F(X(i))$ 
(4)  $p_{\text{best}}(i) = F(X(i))$ 
(5)  $X_{p_{\text{best}}}(i) = X(i)$ 
(6) if  $p_{\text{best}}(i) > g_{\text{best}}$  then
(7)    $g_{\text{best}} = p_{\text{best}}(i)$ 
(8)    $X_{g_{\text{best}}} = X_{p_{\text{best}}}(i)$ 
(9) end if
(10) end for
(11) for  $k=1:N_{\text{max}}$  do
(12)  for  $i=1:M$  do
(13)    生成随机数  $\beta_1, \beta_2$ 
(14)    通过式(14)更新粒子速度  $V(i)$ 
(15)    通过式(15)更新粒子位置  $X(i)$ 
(16)    通过式(16)计算适应度值  $F(X(i))$ 
(17)    if  $F(X(i)) > p_{\text{best}}(i)$  then
(18)       $p_{\text{best}}(i) = F(X(i))$ 
(19)       $X_{p_{\text{best}}}(i) = X(i)$ 
(20)    end if
(21)    if  $p_{\text{best}}(i) > g_{\text{best}}$  then
(22)       $g_{\text{best}} = p_{\text{best}}(i)$ 
(23)       $X_{g_{\text{best}}} = X_{p_{\text{best}}}(i)$ 
(24)    end if
(25)  end for
(26) end for
(27) 通过全局最优粒子位置  $X_{g_{\text{best}}}(i, j)$  得到  $\Omega$  至  $V$ 
    的映射关系  $B$ 
(28) 通过节点映射关系  $B$  及式(2)得到  $\Gamma$  至  $P$  的映
    射关系  $Z$ 
(29) Return  $B: \Omega \rightarrow V, Z: \Gamma \rightarrow P$ 

```

3 仿真结果与分析

本节首先对比基于BPSO算法的路径计算方案与云计算的大数据任务处理时延性能;然后,分析任务分支数据量分配比例及子任务计算复杂度系数的变化对总任务处理时延的影响;最后,比较BPSO算法与其他映射算法应用于路径计算方案时的时延性能。本次仿真采用的计算任务图(DAG)与网络拓扑图(UG)分别如图4、图5所示。

仿真平台采用MATLAB,实验计算机CPU为AMD A8-5550M,内存为4 GB。仿真所用参数均参照文献[18-20]设置,云服务器计算能力 p_c 设为10 GHz,云雾链路的数据传输速率 r_c 设为5 Mb/s,雾

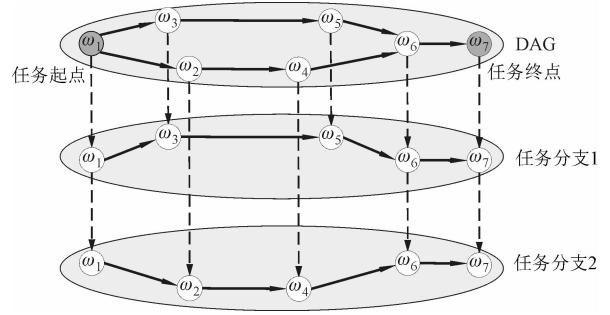


图4 仿真所用计算任务图

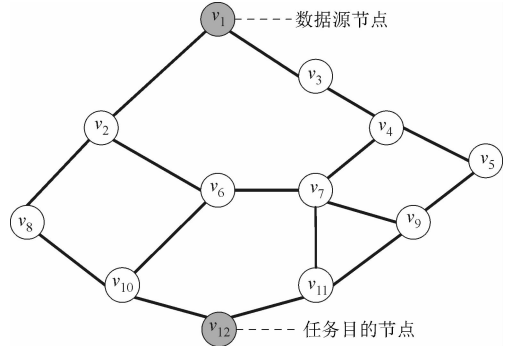


图5 仿真所用网络拓扑图

节点的计算能力 p_{fog} 设为 $[1, 5]$ GHz,雾链路的数据传输速率 r_{fog} 设为 $[10, 20]$ Mb/s,子任务的平均任务计算复杂度系数 α_0 设为 $1\ 900/8$ 。BPSO算法的基本参数为:种群规模 $M=500$,最大迭代次数 $N_{\text{max}}=1\ 000$,加速因子 $\gamma_1=\gamma_2=1$,惯性权重 $\rho=1.5$ 。

3.1 路径计算与云计算时延性能分析对比

为了验证本文所提的基于BPSO算法的路径计算方案的有效性,比较云计算与路径计算方案的大数据任务处理时延性能,并给出不同数据量衰减系数下路径计算方案的时延变化趋势,仿真结果如图6、图7所示。

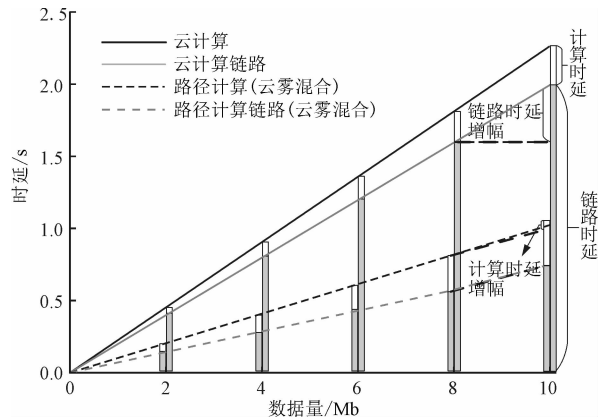


图6 云计算与路径计算方案的大数据任务处理时延对比

在云计算的总时延中链路时延所占比例远远高

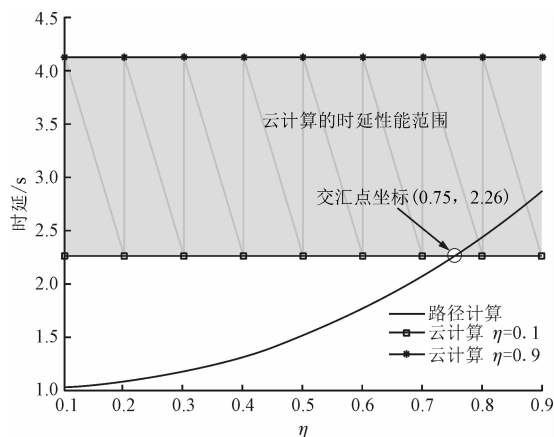


图7 任务数据量为 10 Mb 时数据量衰减系数 η 对总任务处理时延的影响

于节点计算时延,当数据量较小时,云服务器的计算能力可以弥补远距离传输造成的时延损失,但随着数据量增加,链路时延带来的服务响应延迟愈发明显。本文所提路径计算方案基于云雾网络架构实现,雾计算层将原来云端的大数据任务处理过程迁移至网络边缘设备上,在卸载云端计算负荷的同时也降低了数据传输时延。

图6的仿真结果表明:当任务数据处于较小体量时,云计算的处理时延与路径计算相差甚微;随着任务数据量不断增加,云计算的节点计算时延缓慢增长,但其链路时延却显著提高;任务数据量增长 1 Mb 时,云计算链路与计算时延的平均增幅分别为 0.2、0.006 s,同时,路径计算的节点计算时延与链路均保持平缓增长;任务数据量增长 1 Mb 时,路径计算链路与计算时延的平均增幅分别为 0.078、0.031 s。因此,路径计算方案的总处理时延优于云计算。

图7的仿真结果表明,当任务数据量固定时,数据量衰减系数 η 越大,每个子任务计算过程对任务数据量的压缩程度越小,进而导致总任务处理时延较高。用 $\eta=0.1$ 与 $\eta=0.9$ 两条曲线所夹范围表示在 η 变化时云计算的时延性能范围。当 η 从 0.1 开始不断增大时,路径计算任务处理时延也逐步提升,存在一个阈值,使得路径计算达到云计算的时延性能下界。图7中,当任务数据量为 10 Mb 时,此阈值的精确值为 $\eta=0.75$ 。

综上所述,相比云计算,路径计算方案能够有效降低 DAG 形式的医疗大数据任务处理时延,且当数据量衰减系数 η 较小时,路径计算方案更易发挥其时延性能优势。

3.2 总时延与任务分支数据量分配比例的关系

本小节研究任务分支数据量分配比例对总任务处理时延的影响,以任务分支 1 的数据量权重因子 σ_1 为变化量,研究总任务处理时延的变化趋势,仿真结果如图8所示。

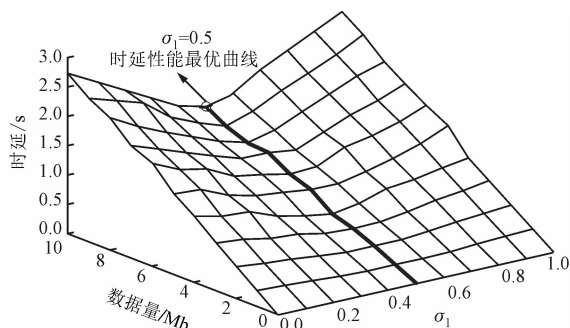


图8 任务分支数据量分配比例对总任务处理时延的影响

图8的仿真结果表明,当 σ_1 由 0 变化至 1 时,总任务处理时延近似以 $\sigma_1=0.5$ 为中心呈对称分布。当 $\sigma_1=0$ 时,小部分网络节点参与了子任务计算,整个雾计算层网络节点的计算能力利用率较低,因此总任务处理时延较高;当 σ_1 逐步增加时,充分利用了雾计算层的计算资源,因此总任务处理时延较小;当 $\sigma_1=0.5$ 时,两任务分支的数据量分配比例相同,且任务处理时延也近似相同,总任务处理时延性能最优。同理, σ_1 由 0.5 变化至 1 时,时延性能表现不断下降。

综上所述,当 DAG 的任务分支数目较多,且各任务分支上的数据量分配均匀时,本文路径计算方案对大数据任务模型支撑能力较好。

3.3 总时延与子任务计算复杂度系数的关系

本小节研究子任务计算复杂度系数对总任务时延的影响。仿真将图5的 DAG 的子任务节点划分为靠近任务起点的前排节点,包含子任务节点 ω_2 、 ω_3 ,及靠近任务终点的后排节点,包含子任务节点 ω_4 、 ω_5 及 ω_6 。DAG 形式的大数据任务以前后排节点的子任务计算复杂度系数为变化参量,可分为前复杂、后简单型 DAG 任务及前简单、后复杂型 DAG 任务。为探究路径计算方案对两种类型的 DAG 任务的支撑能力,仿真为前排节点和后排节点统一设置不同的子任务计算复杂度系数,分别为 α_1 、 α_2 ,且二者满足和为定值 $2\alpha_0$ 的互补关系,以模拟一方简单而另一方复杂的 DAG 任务。仿真以前排节点的子任务计算复杂度系数 α_1 为自变量,以总任务处理时延为因变量,研究子任务计算复杂度系数对总任务时延的影响,仿真结果如图9所示。

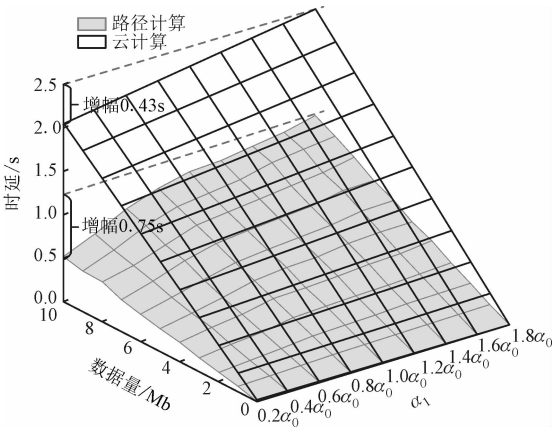


图 9 子任务计算复杂度系数对总任务处理时延的影响

由于子任务会对输入数据量进行压缩,前排节点所需计算的数据量大于后排节点。图 9 的仿真结果表明,当 α_1 较小时,大数据任务为前简单、后复杂的 DAG 任务,大部分任务数据被视为计算复杂度较低的子任务处理,而小部分任务数据却被视为计算复杂度较高的子任务处理,总任务处理时延较小;随着 α_1 不断增大,大数据任务逐渐变为前复杂、后简单的 DAG 任务,总任务处理时延有所提升。由于在云计算的总任务处理时延中,节点计算时延所占比重较小,由计算复杂度系数带来的总时延变化比路径计算小。当任务数据为 10 Mb 且 α_1 由 $0.2\alpha_0$ 递增至 $1.8\alpha_0$ 时,路径计算的处理时延由 0.52 s 增大至 1.27 s,增幅为 0.75 s,而云计算的处理时延由 2.05 s 增大至 2.48 s,增幅为 0.43 s,小于路径计算的时延增长。

3.4 BPSO 算法与其他映射算法的时延性能比较

为了研究 BPSO 算法在路径计算方案中降低大数据任务处理时延的有效性,本小节将 BPSO 算法与随机动态算法(Pick-KX)、加权轮转算法(WRR)、贪婪负载均衡算法(Greedy-LB)等映射算法进行时延性能的比较,对比结果如图 10 所示。

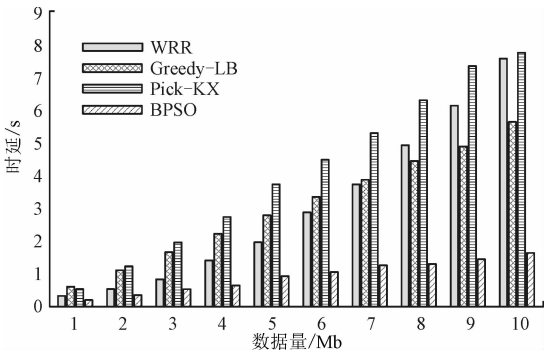


图 10 BPSO 算法与其他映射算法的时延性能对比

图 10 的仿真结果表明,当总数据量小于 3 Mb 时,4 种求解算法得到的时延结果区别甚微,随着数据量不断增大,它们之间的时延性能差距逐渐被表现出来。对于每个子任务而言,Pick-KX 算法随机选取网络节点进行映射,WRR 与 Greedy-LB 算法二者均以网络节点的计算能力及负载情况为依据进行子任务映射,所以其时延性能较 Pick-KX 算法有所提升,但其并未考虑通信时延以及 DAG 中的子任务前后连接关系,仍不是最优;本文采用的 BPSO 算法是以总任务处理时延最小为目标去优化的,综合考虑节点计算时延与通信时延来完成子任务映射,所以其表现出来的时延性能是最优的,在数据量为 10 Mb 时,BPSO 算法相比其他 3 种算法的时延性能均提升 3 倍以上。

综上所述,BPSO 算法比其他 3 种映射算法能更有效地降低路径计算方案的大数据任务处理时延。

4 结 论

针对云计算应用于大数据处理时存在的通信高负荷、任务处理高时延的问题,本文将路径计算技术应用于医疗大数据领域,设计云雾网络架构,并进一步研究了基于 BPSO 算法的任务映射策略。仿真结果表明,本文提出的基于 BPSO 算法的路径计算方案能够有效降低医疗大数据的任务处理时延,增强医疗系统用户体验。然而,本文所考虑的任务映射策略只限于单任务的映射,即将单个 DAG 映射至单个 UG,并未考虑多个 DAG 任务同时映射至单个 UG 的情况;此外,本文所采用的 BPSO 算法隶属智能算法范畴,易陷入局部最优,无法保证每次迭代都可达到最优解。因此,多任务映射及 BPSO 算法求解复杂度优化将是未来的研究重点。

参考文献:

[1] 孟琳, 马金刚, 刘静, 等. 医疗大数据的应用与挑战[J]. 医疗卫生装备, 2018, 39(10): 77-80, 94.
MENG Lin, MA Jingang, LIU Jing, et al. Research and application of big data in medical field [J]. Chinese Medical Equipment Journal, 2018, 39(10): 77-80, 94.
[2] 张宇晴, 常志远, 季建菲. 应用大数据推动“互联网+医疗”发展问题研究 [J]. 时代金融, 2018, 718(36): 209, 212.
ZHANG Yuqing, CHANG Zhiyuan, JI Jianfei. Using big data to promote research on “Internet + medical” development [J]. Times Finance, 2018, 718(36):

- 209, 212.
- [3] CHEN T, CHEN Y, CHEN J, et al. Flu trend prediction based on massive data analysis [C] // 2018 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis. Piscataway, NJ, USA: IEEE, 2018: 304-308.
- [4] ARCHENAA J, ANITA E A M. A survey of big data analytics in healthcare and government [J]. *Procedia Computer Science*, 2015, 50: 408-413.
- [5] 舒影岚, 陈艳萍, 吉臻宇, 等. 健康医疗大数据研究进展 [J]. *中国医学装备*, 2019, 16(1): 149-153.
- SHU Yinglan, CHEN Yanping, JI Zhenyu, et al. Research progress of big data of healthcare and medical treatment [J]. *China Medical Equipment*, 2019, 16(1): 149-153.
- [6] 张世红, 史森, 杨小冉. 健康医疗大数据应用面临的挑战及策略探讨 [J]. *中国卫生信息管理杂志*, 2018, 15(6): 629-632, 658.
- ZHANG Shihong, SHI Sen, YANG Xiaoran. Discussion on the challenges and strategies of healthcare big data application [J]. *Chinese Journal of Health Informatics and Management*, 2018, 15(6): 629-632, 658.
- [7] 过骏愈. 基于云计算和大数据的医院信息化建设 [J]. *电子技术与软件工程*, 2019, 150(4): 195.
- GUO Junyu. Based on cloud computing and big data of hospital informatization construction [J]. *Electronic Technology & Software Engineering*, 2019, 150(4): 195.
- [8] VUKOBRATOVIC D, JAKOVETIC D, SKACHEK V, et al. CONDENSE: a reconfigurable knowledge acquisition architecture for future 5G IoT [J]. *IEEE Access*, 2016, 4: 3360-3378.
- [9] GALLAGER R. Finding parity in a simple broadcast network [J]. *IEEE Transactions on Information Theory*, 1988, 34(2): 176-180.
- [10] KUSHILEVITZ E, MANSOUR Y. Computation in noisy radio networks [C] // *Proceedings of the 1998 9th Annual ACM SIAM Symposium on Discrete Algorithms*. Philadelphia, PA, USA: SIAM, 1998: 236-243.
- [11] FEIGE U, KILIAN J. Finding OR in a noisy broadcast network [J]. *Information Processing Letters*, 2000, 73(1/2): 69-75.
- [12] 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.
- [13] SHUKLA S, VYAVAHARE P, KURI J, et al. Fast arbitrary function computation over a wireless network: a linear programming [C] // *IEEE Wireless Communications and Networking Conference*. Piscataway, NJ, USA: IEEE, 2015: 2091-2096.
- [14] VUKOBRATOVIC D. Network function computation as a service in future 5G machine type communications [C] // *9th International Symposium on Turbo Codes and Iterative Information Processing*. Piscataway, NJ, USA: IEEE, 2016: 365-369.
- [15] KAMINSKI N, MACALUSO I, DI PASCALE E, et al. A neural-network-based realization of in-network computation for the internet of things [C] // *2017 IEEE International Conference on Communications*. Piscataway, NJ, USA: IEEE, 2017: 7996821.
- [16] 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 Xiuli, REN Zhiyuan, SHI Chenhua, 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, 51(10): 71-77.
- [17] VYAVAHARE P, SHETTY A. On optimal embedding for distributed computation of arbitrary functions [C] // *10th International Conference on Signal Processing and Communications*. Piscataway, NJ, USA: IEEE, 2014: 6984006.
- [18] HOU X, REN Z, CHENG W, et al. Fog based computation offloading for swarm of drones [C] // *2019 IEEE International Conference on Communications*. Piscataway, NJ, USA: IEEE, 2019: 8761932.
- [19] HASSAN M A, XIAO Mengbai, WEI Qi, et al. Help your mobile applications with fog computing [C] // *12th Annual IEEE International Conference on Sensing, Communication and Networking*. Piscataway, NJ, USA: IEEE, 2015: 49-54.
- [20] OLGA M, PASCUAL-ISERTE A, VIDAL J. Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading [J]. *IEEE Transactions on Vehicular Technology*, 2014, 64(10): 4738-4755.
- [21] THINH T Q, TANG Jianhua, LA Q D, et al. Offloading in mobile edge computing: task allocation and computational frequency scaling [J]. *IEEE Transactions on Communications*, 2017, 65(8): 3571-3584.

(编辑 刘杨)