

## 流特征感知的软件定义网络控制器动态关联机制

张少军\* 兰巨龙 江逸茗 孙鹏浩

(国家数字交换系统工程技术研究中心 郑州 450002)

**摘要:** 在部署分布式控制平面的软件定义网络中, 控制器与交换机的关联仅以数据流请求的数量分布作为控制资源分配的依据。该文针对这一问题, 以数据流的源目的地址特征为例, 对不同特征数据流的控制资源消耗进行了分析, 提出在控制资源分配中应对数据流的特征分布加以考虑。然后, 设计了一种流特征感知的控制器关联决策机制, 并针对网络流的动态变化特性设计了一种快速求解算法。仿真结果表明, 与基于负载均衡的机制对比, 所提机制在使用模拟退火算法求解时能节省10%~20%的控制资源消耗; 所提快速求解算法可节省10%的资源消耗, 且相比模拟退火算法具有较大的速度优势和良好的可扩展性。

**关键词:** 软件定义网络; 控制器关联; 资源分配; 流特征感知

中图分类号: TP393.2

文献标识码: A

文章编号: 1009-5896(2018)09-2050-07

DOI: 10.11999/JEIT171149

## Flow Characteristics Aware Dynamic Controller Assignment in Software-defined Networking

ZHANG Shaojun LAN Julong JIANG Yiming SUN Penghao

(National Digital Switching System Engineering and Technological R & D Center, Zhengzhou 450002, China)

**Abstract:** In Software-Defined Networking (SDN) with distributed control plane, the switches are assigned to controllers using only the quantity distribution of flow requests as the basis of resource allocation. To address this issue, the control resource consumption of flow requests processing with different characteristics is analyzed taking the source and destination of flow as an example, from which a conclusion is drawn that the characteristics distribution of flow should be taken into account when allocating control resource. Then, a flow characteristics aware controller assignment model is designed, and a fast algorithm coping with the fluctuation of flow request is proposed. Simulation results show that when solving with the simulated annealing algorithm, the model can save 10%~20% of control resource compared with the load balancing model; with 10% of resource saving, the proposed algorithm outperforms the simulated annealing algorithm in execution speed and scalability.

**Key words:** Software-Defined Networking (SDN); Controller assignment; Resource allocation; Flow characteristics aware

### 1 引言

互联网封闭僵化的体系结构已成为其演进和创新的障碍。软件定义网络(Software-Defined Networking, SDN)<sup>[1]</sup>将控制平面与数据平面分离, 通过逻辑集中的控制平面和开放可编程的数据平面, 为网络提供了灵活的配置和管理, 方便了新协

议的开发和部署。为了提供高性能、可扩展的控制服务<sup>[2]</sup>, SDN采用分布式控制平面<sup>[3]</sup>、部署多个控制器对数据平面进行分域管理, 每个控制器与其管理域内所有交换机相关联, 控制器之间相互协作实现高效的网络控制。

SDN通过控制平面的计算、存储、带宽等控制资源来处理网络事件。为了改善控制平面的吞吐量、响应速度并提高其资源利用率, 需要对控制资源进行优化分配, 调节其在网络中的分布, 使其与数据平面的需求分布相契合。目前, 分布式控制平面中, 控制资源按控制器粒度分布在网络中, 控制资源优化分配主要通过控制器优化部署<sup>[4]</sup>和控制器动态关联<sup>[5]</sup>来实现。其中, 控制器优化部署主要根

收稿日期: 2017-12-06; 改回日期: 2018-05-18; 网络出版: 2018-07-12

\*通信作者: 张少军 zhangsj07@126.com

基金项目: 国家自然科学基金(61521003, 61572519, 61502530), 国家863计划项目(2015AA016102)

Foundation Items: The National Natural Science Foundation of China (61521003, 61572519, 61502530), The National 863 Program of China (2015AA016102)

据数据平面中各个交换机所产生流请求的空间分布确定控制资源分布。文献[6]在分布式控制平面中设计了一种控制器部署和关联算法,但是该算法中,控制负载的衡量指标是交换机的数目。实际上,由于每个交换机上连接的主机功能不同,不同交换机产生的控制负载也不相同,以交换机产生的流请求数目来衡量控制负载更为合理。文献[4,7]在研究控制器放置方案时均以交换机的流请求数目作为控制负载的衡量标准。控制器动态关联基于网络中流请求分布的时变现实<sup>[8]</sup>,根据流请求的数量分布动态调整控制器与交换机的关联关系,实现控制平面负载均衡。基于OpenFlow<sup>[9]</sup>的多控制器机制,文献[5]提出一种交换机迁移协议来动态调整控制器与交换机的关联关系。在此基础上,文献[10]和文献[11,12]分别以控制平面负载均衡和最小化控制平面响应时间为目标,根据数据平面流请求的数量分布,设计了不同的动态控制器关联机制。

SDN网络中,数据流作为SDN各种网络应用的承载实体,不同的数据流具有不同的特征,控制平面处理不同特征的流请求时耗费的控制资源也不尽相同。最典型的流特征是数据流的源、目的地址信息,当一个数据流的源地址和目的地址位于不同的控制域时,对其进行处理所需的控制资源要高于二者位于同一控制域的情况。然而,现有的控制资源分配机制仅仅以流请求的数量分布作为数据平面的需求依据,为流请求高的交换机分配较多、较近的控制资源,而忽略了流请求的特征分布。这类机制可能导致相互之间流请求较多的交换机被分配在不同的控制域中,增加控制平面的资源消耗,降低其响应能力。因此,在SDN网络中,依据流请求的特征分布进行控制资源的优化分配,对提升控制平面性能和资源利用率具有重要的意义。文献[13]在设计控制器部署方案时指出应考虑流请求的特征分布,但是这里的特征指的是数据流的源交换机信息,该信息实际上与流请求的数量分布等价。文献[8]分析了校园网、企业网及数据中心网络的流量特征,特别关注了机架内流量和机架间流量,指出对应用进行优化部署能极大地减少机架间流量。这一论断也是本文工作的启发。

综上,本文提出一种流特征感知的SDN控制资源优化分配机制。具体而言,本文根据SDN网络中流请求的源目的地址信息的动态分布,设计了一种控制器动态关联机制,通过动态调整控制器与交换机的关联关系,使控制资源分布与数据平面中流请求的特征分布相契合,从而降低控制资源消耗,提高资源利用率。本文的贡献在于:(1)根据数据流

的源目的地址信息对不同流请求带来的控制负载进行深入分析,指出在控制资源优化分配中须对流特征加以考虑;(2)以最小化控制资源消耗为目标设计了一种控制器关联决策机制;(3)为实现动态控制器关联,基于排序不等式原理设计了一种次优的快速求解算法。

## 2 流特征与控制资源消耗分析

SDN网络中,数据流是各种网络应用的承载实体,包头信息是流特征的直接体现。路径计算作为SDN最普遍的应用,其所处理的流转发请求也是控制平面最主要的负载,而处理流转发请求时,涉及到的流特征主要指数据流的源目的地址信息。因此,本节以数据流的源目的地址信息为特征,对处理不同特征流请求所消耗的控制资源进行分析。

### 2.1 流特征分类

如图1所示,在部署分布式控制平面的SDN网络中,各个控制器对数据平面进行分域管理,每个控制器仅负责处理各自域内的网络事件,多个控制器协作进行全局网络事件的处理。网络中的数据流可分为域内流和跨域流两类。

**定义1** 域内流:源地址和目的地址均位于同一个控制域的数据流。如图1中源主机到目的主机1的数据流。

**定义2** 跨域流:源地址和目的地址分别位于不同控制域的数据流。如图1中源主机到目的主机2的数据流。

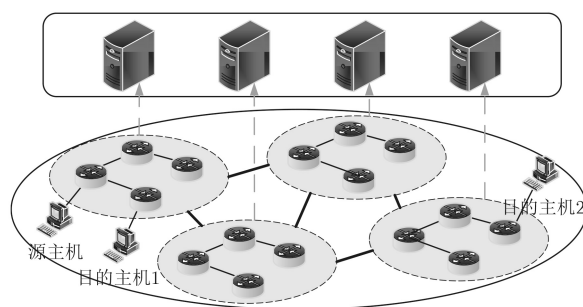


图1 部署分布式控制平面的SDN网络

控制平面对域内流和跨域流的处理流程不同,消耗的资源也不同。下面结合处理流程对二者的控制资源消耗展开分析。

### 2.2 控制资源消耗分析

对域内流,控制平面的处理流程与单个控制器对转发请求的处理流程相同,如下所示:

(1)当数据流的第1个数据包到达交换机时,若交换机中没有匹配的流表项,则上传转发请求(如OpenFlow的PACKET\_IN消息)到本域控制器;

(2) 控制器根据网络视图计算转发路径, 生成流表项;

(3) 控制器将流表项下发给路径涉及的交换机。对跨域流, 控制平面的处理流程如下所示:

(1) 控制器收到跨域转发请求, 根据网络视图计算全网转发路径, 获得数据包应流向的下一控制域;

(2) 控制器计算发往下一控制域的域内转发路径, 生成并下发流表项;

(3) 下一控制域的控制器收到转发请求, 若目的地址不在本域, 则按照步骤(1)和步骤(2)处理, 否则按域内流的处理流程处理。

跨域流的处理涉及该数据流流经的所有控制域, 对此给出如下定义:

**定义3** 控制器图: 以各个控制域为节点, 控制域之间的链路为边所得的图。控制器图是将网络拓扑中控制域抽象为虚拟交换机后得到的抽象拓扑。

**定义4** 控制路径: 以数据流经过的所有控制域为节点, 数据流路径中的域间链路为边组成的路径。控制路径是数据流在控制器图中对应的路径。相应地, 控制路径长度 $p_C$ 表示数据流流经的控制域的个数。显然, 对域内流,  $p_C = 1$ 。

控制平面在处理流请求时, 需要消耗计算、带宽和存储等控制资源。其中存储资源主要用于存储网络视图和状态信息, 是控制器的基础消耗, 且资源成本较低, 因此不作考虑; 计算资源主要用于响应流请求和计算转发规则; 带宽资源主要用于传输控制信令, 包括流请求上传和流表项下发等。由于控制平面对跨域流的处理可分为 $p_C$ 个域内流处理流程, 跨域流的计算和带宽资源消耗均相当于 $p_C$ 个域内流的计算和带宽资源消耗。因此, 跨域流的控制资源消耗可建模为 $p_C$ 个域内流的控制资源消耗。

根据上述分析, 在分布式控制平面中, 通过调整控制器与交换机的关联关系, 降低SDN网络中跨域流的控制资源消耗, 可以有效降低网络整体的控制资源消耗。以此为例, 在其他SDN应用中, 同样可以对相关流特征的控制资源消耗进行量化分析, 综合考虑流的数量分布和特征分布来确定控制资源的分配, 从而降低控制资源的消耗。

### 3 问题建模

本节对流特征感知的控制器映射问题(Flow-characteristics Aware Controller Assignment, FACA)进行建模。具体而言, 是针对数据流的源目的地址特征, 根据域内流和跨域流的控制资源消耗, 通过动态调整控制器与交换机的关联关系, 来

降低控制平面处理转发请求的资源消耗。

给定一个SDN网络 $G = (S, E)$ , 其中包含 $N$ 个交换机 $S = \{s_1, s_2, \dots, s_N\}$ 和 $M$ 个控制器 $C = \{c_1, c_2, \dots, c_M\}$ 。控制器 $c_m$ 最多可处理的转发请求数即控制器容量为 $a_m$ 。控制器与交换机之间使用带内控制通道维护连通性, 并使用最短路径通信。控制器 $c_m$ 与交换机 $s_n$ 的最短路径距离为 $d_{mn}$ , 考虑响应时间约束, 该距离应小于阈值 $\delta$ 。

数据平面的流请求分布随时间变化, 以下对流请求分布和控制器关联的描述均表示时隙 $t$ 的情况, 为表述简单, 省略参数 $t$ 。

定义交换机 $s_i$ 到 $s_j$ 的流请求速率为单位时间内 $s_i$ 向控制器提交的、以 $s_j$ 上所有主机为目的地址的流请求的总和, 设为 $f_{ij}$ , 其中 $i, j = 1, 2, \dots, N$ 。据此可得, 交换机 $s_i$ 的流请求速率为 $\lambda_i = \sum_{j=1}^N f_{ij}$ 。

定义控制器关联函数为 $m = \varphi(n)$ , 表示交换机 $s_n$ 关联到控制器 $c_m$ 。控制器关联关系可同样定义为矩阵 $\mathbf{X} = [x_{nm}]_{N \times M}$ , 当 $s_n$ 关联到 $c_m$ 时 $x_{nm}$ 为1, 否则为0。据此可得,  $c_m$ 的控制负载为 $\theta_m = \sum_{n=1}^N x_{nm} \lambda_n$ , 考虑控制器容量约束, 应满足 $\theta_m < a_m$ 。

根据2.2节的分析, 跨域流的控制资源消耗相当于 $p_C$ 个域内流的控制资源消耗。实际场景中, 对不同的域内流和不同的控制域, 处理转发请求所消耗的控制资源不尽相同。为简化分析, 这里采用平均化的思路, 忽略域内流资源消耗的差异, 采用控制路径长度来衡量转发请求消耗的控制资源。对于源交换机 $s_i$ 到目的交换机 $s_j$ 的数据流, 控制路径长度表示为 $p_{\varphi(i)\varphi(j)}$ , 控制资源消耗为 $f_{ij} p_{\varphi(i)\varphi(j)}$ 。

综上, FACA问题可以定义为

$$\min \sum_{i=1}^N \sum_{j=1}^N f_{ij} p_{\varphi(i)\varphi(j)} \quad (1)$$

$$\text{s.t. } \theta_m < a_m, \forall m \quad (2)$$

$$d_{mn} < \delta, \forall m, n \quad (3)$$

$$x_{nm} \in \{0, 1\}, \forall m, n \quad (4)$$

$$\sum_{m=1}^M x_{nm} = 1, \forall n \quad (5)$$

$$x_{nm} = \begin{cases} 1, & \varphi(n) = m \\ 0, & \text{其他} \end{cases} \quad (6)$$

式(1)是该问题的优化目标, 即最小化控制平面处理转发请求的资源消耗。式(2)和式(3)分别代

表控制器容量约束以及交换机和控制器的距离约束。式(5)表示每个交换机只能关联到一个控制器。式(6)表示控制器关联函数和关联矩阵之间的等价关系。

## 4 问题求解

### 4.1 问题分析

分别定义交换机之间流请求速率和控制域之间资源消耗的矩阵形式，即 $\mathbf{F}=[f_{ij}]_{N \times N}$ ， $\mathbf{P}=[p_{hk}]_{M \times M}$ ，FACA问题的目标函数可改写为矩阵形式 $\text{tr}(\mathbf{X}\mathbf{P}\mathbf{X}^T\mathbf{F}^T)$ 。可以看出，该目标函数是二次型的。实际上，将问题简化，假设每个交换机产生的负载相同，则每个控制器关联的交换机数目是确定的，设为 $K_m$ ，将每个控制器分为 $K_m$ 个控制器切片，每个切片关联一个交换机，此时FACA问题等价于二次分配问题<sup>[14]</sup>(Quadratic Assignment Problem, QAP)。QAP问题是经典的组合优化问题，已被证明是NP难。目前，对QAP问题多采用模拟退火算法<sup>[15]</sup>等启发式方法来求取近似解。

根据上述分析，FACA问题也是NP难问题。精确求解只能依靠遍历，但是遍历的复杂度为 $M^N$ ，在问题规模较大时时间复杂度过高。类似于QAP问题，FACA问题也可以采用模拟退火算法获取近似最优的解。但是网络中流请求分布动态随机变化，模拟退火算法作为一种随机搜索算法，其求解速度不能满足控制器动态关联的快速性要求，无法在网络运行时展开决策。因此，有必要设计一种适用于运行时的快速控制器动态关联算法。

### 4.2 快速求解算法

为实现FACA问题的快速求解，本文提出一种排序关联算法(Sequence Assignment, SeqAsn)。该算法的理论依据是排序不等式原理<sup>[16]</sup>(sequence inequality)，如下所示：

**定理1** 排序不等式原理。设 $a_1 \leq a_2 \leq \dots \leq a_n$ 与 $b_1 \leq b_2 \leq \dots \leq b_n$ 为两组实数， $x_1, x_2, \dots, x_n$ 是 $b_1, b_2, \dots, b_n$ 的任意排列，则

$$a_1 b_n + a_2 b_{n-1} + \dots + a_n b_1 \leq a_1 x_1 + a_2 x_2 + \dots + a_n x_n \leq a_1 b_1 + a_2 b_2 + \dots + a_n b_n$$

在流特征感知的控制器关联问题中，以序列 $a_1 \leq a_2 \leq \dots \leq a_n$ 代表交换机对之间的数据流 $f_{ij}$ 按速率从小到大排列，序列 $b_1 \leq b_2 \leq \dots \leq b_n$ 代表控制器对之间控制路径长度 $p_{hk}$ 从小到大排列。基于排序不等式原理，为了最小化控制资源的消耗，最直观的思路是：交换机对之间的数据流速率越大，其消耗的控制资源应越少，即流请求速率越大的一对交换机应分别关联到控制路径越短的一对控制器上。因此，排序关联算法的核心思想是，调整控制

表1 排序关联算法

#### 算法1 排序关联算法

输入：交换机间流请求速率矩阵 $\mathbf{F}=[f_{ij}]_{N \times N}$ ；

控制域间资源消耗矩阵 $\mathbf{P}=[p_{hk}]_{M \times M}$ ；

控制器容量 $a_m$ ；

控制器与交换机之间的距离阈值 $\delta$ ；

输出：控制器与交换机的关联关系 $\varphi(\cdot)$

- (1) 对每个交换机，根据距离约束公式(3)确定候选控制器集合 $\Gamma(s_i)$ ；
- (2) 将交换机对 $(s_i, s_j)$ 按 $f_{ij}$ 降序排列，形成序列 $L_S$ ；
- (3) 初始化已关联交换机集合 $\Omega = \emptyset$ ；
- (4) **for**  $(s_i, s_j) \in L_S$  **do**
- (5) **if**  $s_i \notin \Omega$  **or**  $s_j \notin \Omega$  **then** /\*若2个交换机均已关联，则跳过\*/
- (6) 获取 $\Gamma(s_i)$ 与 $\Gamma(s_j)$ ， $(s_i, s_j)$ 的候选控制器对 $(c_h, c_k)$ 的集合为直积 $\Gamma(s_i) \times \Gamma(s_j)$ ；
- (7) 将候选控制器对按 $p_{hk}$ 升序排列，形成序列 $L_C^i$ ；
- (8) **for**  $(c_h, c_k) \in L_C^i$  **do**
- (9) 将 $(s_i, s_j)$ 关联到 $(c_h, c_k)$ ；
- (10) **if**  $\theta_h < a_n$  **and**  $\theta_k < a_k$  **then** /\*若关联后有控制器超载，则跳过\*/
- (11)  $\varphi(s_i) = c_h, \varphi(s_j) = c_k$ ; /\*控制器接受关联\*/
- (12)  $\Omega = \Omega \cup \{s_i, s_j\}$ ；
- (13)  $\Gamma(s_i) = \{\varphi(s_i)\}, \Gamma(s_j) = \{\varphi(s_j)\}$ ；
- (14) **goto** step 4;
- (15) **end if**
- (16) **end for**
- (17) **end if**
- (18) **end for**

器与交换机的关联关系，使交换机之间控制路径分布与交换机之间数据流分布的特征图样反向对齐，从而减少高成本跨域通信的场景。

算法执行过程中，首先对每个交换机根据距离约束公式(3)确定其候选控制器集合，将距离约束条件转化为缩小的解空间；然后，将所有的交换机源目的对按流请求速率降序排序；之后遍历交换机对，获取其候选控制器对的集合，将候选控制器对按控制路径长度升序排序；而后逐对尝试，控制器无超载现象时完成关联。对每对交换机依次执行，直至关联完毕。

步骤13中，若交换机已关联，其候选控制器集合将仅包含其关联的控制器。步骤7对控制器对排序时，若两个控制器对的 $p_{hk}$ 相同，排序时要按照控制器剩余容量从大到小排序，从而尽可能防止负载集中到部分控制器，造成控制器超载。

### 4.3 算法复杂度分析

SeqAsn算法锁定搜索空间和搜索方向为确定

的序列, 搜索空间较小。对包含 $N$ 个交换机和 $M$ 个控制器的SDN网络, SeqAsn算法首先对交换机对进行排序, 排序规模为 $N^2$ , 采用稳定排序算法冒泡排序的时间复杂度为 $O(N^4)$ 。在对每个交换机对进行关联决策时, 候选控制器对的集合规模最大为 $M^2$ , 采用冒泡排序的时间复杂度为 $O(M^4)$ , 遍历控制器对集合进行关联决策的时间复杂度最大为 $O(M^2)$ 。因此, SeqAsn算法的时间复杂度为 $O(N^4 + N^2M^4 + N^2M^2)$ , 避免了网络规模较大时的指数爆炸现象。

## 5 仿真分析

### 5.1 仿真环境搭建

为了说明FACA机制的性能, 本节将其与文献[10]中的控制平面负载均衡机制(Load Balancing for Distributed Control plane, LBDC)做比较, 对比二者的控制资源消耗。LBDC的目的是通过交换机迁移来实现控制平面的负载均衡。同时, 为了说明SeqAsn算法的求解性能, 本节采用模拟退火算法(Simulated Annealing, SimAnn)对FACA问题的求解作为参照, 并对比二者的求解结果和运行时间。模拟退火算法的初始解设为LBDC解得的控制器关联关系, 通过随机选择交换机并在其候选控制器集合中随机选择控制器来获得新解。模拟退火算法的其他参数设置如表2所示。

表2 模拟退火算法的参数设置

初始温度	终止温度	迭代次数	降温速率	接受准则
1000	1e-3	100	0.95	Metropolis准则

本节首先在Internet Topology Zoo提供的公共拓扑中挑选了5个具有显著规模差异的网络拓扑展开仿真实验, 这些拓扑的网络规模及相关控制域参数如表3所示。实验过程中, 为了模拟数据平面成对流分布的空间不均匀特性, 交换机之间的流请求速率按照均值为5、方差为1的正态分布随机产生。控制器容量根据这一流请求速率和网络规模进行设置。最后, 采用跳数作为实验中设备间距离的衡量标准, 且控制路径由最短路径算法获得。

### 5.2 仿真分析

在5个拓扑中, 分别运行LBDC、SeqAsn和SimAnn算法, 对其控制资源消耗进行对比。将仿真结果分别按照各拓扑中LBDC算法的结果进行归一化, 得到图2所示的控制资源消耗对比结果。结果显示, 相比负载均衡机制, 采用FACA机制来设计控制器与交换机的关联关系, 并使用SimAnn算法进行求解, 能减少10%到20%的控制资源消耗。Se-

表3 实验拓扑数据

拓扑	节点数	边数	控制器个数	控制器容量	距离阈值
RedIris	19	32	4	600	3
GEANT2009	34	52	6	1800	3
DFN	58	87	8	3000	4
Interoute	110	159	10	8000	5
TATANld	145	194	12	12000	7

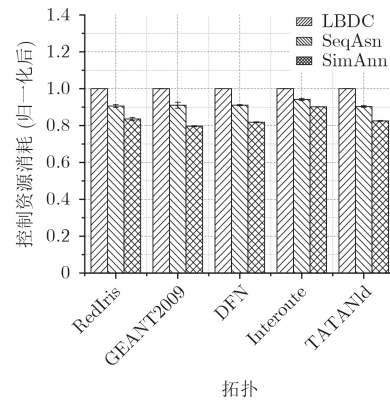


图2 控制资源消耗对比

qAsn算法是一种贪婪算法, 它以交换机对为中心进行控制器对的搜索, 其求解结果不如SimAnn算法。即便如此, SeqAsn算法依然能减少10%左右的控制资源消耗。

SeqAsn算法与SimAnn算法在各拓扑中的平均运行时间如表4所示。可以看出, 虽然SeqAsn算法的精度不如SimAnn算法, 但是其运行速度具有较大的优势, 适用于在网络运行时展开控制器动态关联决策。此外, SeqAsn算法的求解时间不会随着网络规模的扩大而产生爆炸式增长, 具有良好的可扩展性。SimAnn算法精度高但消耗时间较长, 适用于网络初始部署时进行控制器关联决策, 或者流分布变化缓慢的网络。

由于跨域流消耗的控制资源较多, FACA机制在确定控制器关联时会倾向于将尽可能多的交换机集中关联至少数个控制器, 以减少跨域流的数目。这种倾向会造成控制平面负载失衡, 降低控制平面的响应速度, 甚至造成部分控制器因超载而瘫

表4 算法运行时间(ms)

拓扑	SeqAsn	SimAnn
RedIris	6.5	78.36
GEANT2009	8.9	449.68
DFN	11.68	841.06
Interoute	18.32	2223.36
TATANld	25.84	4355.86

疾。图3所示为LBDC, SeqAsn和SimAnn这三种算法的控制平面负载方差, 仿真结果按照LBDC算法的结果进行归一化。可以看出, SimAnn算法在追求最少的控制资源消耗的同时, 会带来较严重的负载失衡问题。而SeqAsn算法在设计时就兼顾了负载均衡因素, 在进行交换机分配时会同时考虑将其关联到剩余容量较多的控制器, 因此其能够提供较好的负载均衡性能。

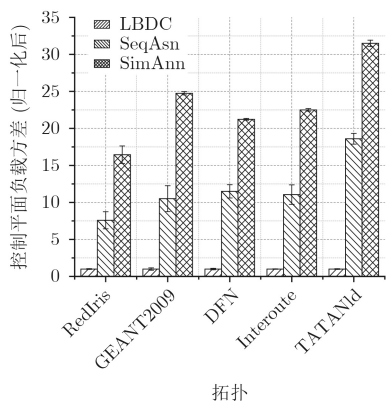


图3 控制平面负载均衡程度对比

最后, 为了进一步分析FACA机制和SeqAsn算

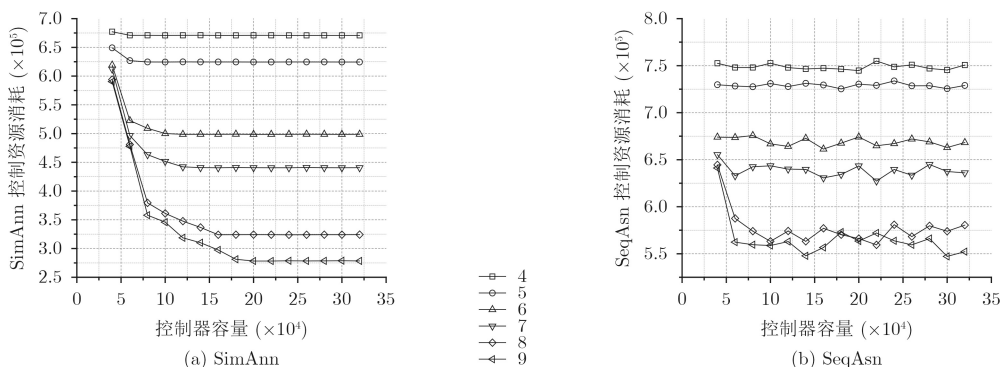


图4 控制器容量和控制距离阈值的影响

法, 本节使用GT-ITM生成网络规模为250的模拟网络拓扑, 在其中部署16个控制器, 评估控制器容量和控制距离阈值对算法性能的影响。SimAnn算法和SeqAsn算法的仿真结果分别如图4(a)和图4(b)所示, 其中控制距离阈值(4, 5, 6, 7, 8, 9)的影响由多条曲线表示。SimAnn算法由于采用随机搜索方法, 可以大致反映FACA机制的情况。从图4(a)可以看出, 采用SimAnn算法得到的控制资源消耗随着控制器容量和控制器可覆盖范围的增长而降低, 这是由于随着控制器容量的提升以及控制器可覆盖范围的扩大, 单个控制器可关联的交换机增加, 使得跨域流的数量减少, 整体的控制资源消耗随之降低。另外, 控制器容量和可覆盖范围的影响会互相制约, 因此在进行控制器部署时, 可以根据这一制约关系选择合适的控制器容量和控制范围阈值, 避免不必要的投入。如图4(b)所示, SeqAsn算法得到的控制资源消耗同样会随着控制器可覆盖范围的增长而降低。然而, 由于SeqAsn算法以交换机对为中心进行控制器对的搜索, 并在选择控制器时综合考虑了负载均衡因素, 因此控制器容量对算法结果的影响不大。

## 6 结束语

本文在分布式控制平面的控制资源优化分配问题中, 根据不同特征的数据流消耗的控制资源不同这一分析, 提出一种流特征感知的控制器动态关联机制, 为控制平面资源分配提供更加深入和准确的决策依据。本文以数据流的源目的地址特征为例, 结合分布式控制平面中域内流和跨域流对控制资源消耗的不同, 以最小化控制资源消耗为目标设计了一种控制器关联决策机制。最后, 为实现运行时的动态控制器关联, 本文基于排序不等式原理设计了一种次优的快速求解算法SeqAsn。实验结果表明, 流特征感知的控制器关联机制能有效减少SDN运行

中的控制资源消耗; SeqAsn算法能够实现该问题的快速求解, 从而应用于控制器动态关联决策。本文的下一步工作是设计一种动态拓扑机制, 通过减小控制路径长度来降低跨域场景的控制资源消耗。

## 参考文献

- [1] KREUTZ D, RAMOS F M V, VERISSIMO P E, *et al.* Software-defined networking: A comprehensive survey[J]. *Proceedings of the IEEE*, 2015, 103(1): 14-76. doi: 10.1109/JPROC.2014.2371999.
- [2] KARAKUS M and DURRESI A. A survey: Control plane scalability issues and approaches in Software-Defined Networking (SDN)[J]. *Computer Networks*, 2017, 112:

- 279–293. doi: [10.1016/j.comnet.2016.11.017](https://doi.org/10.1016/j.comnet.2016.11.017).
- [3] BERDE P, GEROLA M, HART J, *et al.* ONOS: towards an open, distributed SDN OS[C]. Hot Topics in Software Defined Networking (HotSDN), Chicago, USA, 2014: 1–6.
- [4] LIAO Jianxin, SUN Haifeng, WANG Jingyu, *et al.* Density cluster based approach for controller placement problem in large-scale software defined networkings[J]. *Computer Networks*, 2017, 112: 24–35. doi: [10.1016/j.comnet.2016.10.014](https://doi.org/10.1016/j.comnet.2016.10.014).
- [5] DIXIT A, HAO F, MUKHERJEE S, *et al.* Towards an elastic distributed SDN controller[J]. *ACM SIGCOMM Computer Communication Review*, 2013, 43(4): 7–12. doi: [10.1145/2534169.2491193](https://doi.org/10.1145/2534169.2491193).
- [6] JIMENEZ Y, CERVELLO-PASTOR C, and Garcia A J. On the controller placement for designing a distributed SDN control layer[C]. IFIP Networking Conference, Trondheim, Norway, 2014: 1–9.
- [7] 伊鹏, 刘邦舟, 王文博, 等. 一种考虑软件定义网络控制节点故障的控制器部署和交换机迁移方法[J]. 电子与信息学报, 2017, 39(8): 1972–1978. doi: [10.11999/JEIT161216](https://doi.org/10.11999/JEIT161216).  
YI Peng, LIU Bangzhou, WANG Wenbo, *et al.* Controller placement and switch immigration strategy for SDN controller failure[J]. *Journal of Electronics & Information Technology*, 2017, 39(8): 1972–1978. doi: [10.11999/JEIT161216](https://doi.org/10.11999/JEIT161216).
- [8] BENSON T, AKELLA A, and MALTZ D. Network traffic characteristics of data centers in the wild[C]. ACM Internet Measurement Conference (IMC), Melbourne, Australia, 2010: 267–280.
- [9] MCKEOWN N, ANDERSON T, BALAKRISHNAN H, *et al.* OpenFlow: Enabling innovation in campus networks[J]. *ACM SIGCOMM Computer Communication Review*, 2008, 38(2): 69–74. doi: [10.1145/1355734.1355746](https://doi.org/10.1145/1355734.1355746).
- [10] GAO Xiaofeng, KONG Linghe, LI Weichen, *et al.* Traffic load balancing schemes for devolved controllers in mega data centers[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2017, 28(2): 572–585. doi: [10.1109/TPDS.2016.2579622](https://doi.org/10.1109/TPDS.2016.2579622).
- [11] WANG Tao, LIU Fangming, GUO Jian, *et al.* Dynamic SDN controller assignment in data center networks: Stable matching with transfers[C]. IEEE International Conference on Computer Communications (INFOCOM), San Francisco, USA, 2016: 1–9.
- [12] WANG Tao, LIU Fangming, and XU Hong. An efficient online algorithm for dynamic SDN controller assignment in data center networks[J]. *IEEE/ACM Transactions on Networking*, 2017, 25(5): 2788–2801. doi: [10.1109/TNET.2017.2711641](https://doi.org/10.1109/TNET.2017.2711641).
- [13] SOLIMAN M, NANDY B, LAMBADARIS I, *et al.* Source routed forwarding with software defined control, considerations and implications[C]. International Conference on emerging Networking EXperiments & Technologies (CoNEXT), Nice, France, 2012: 43–44.
- [14] LOIOLA E M, ABREU N M M D, BOAVENTURA-NETTO P O, *et al.* A survey for the quadratic assignment problem[J]. *European Journal of Operational Research*, 2007, 176(2): 657–690. doi: [10.1016/j.ejor.2005.09.032](https://doi.org/10.1016/j.ejor.2005.09.032).
- [15] KIRKPATRICK S. Optimization by simulated annealing: quantitative studies[J]. *Journal of Statistical Physics*, 1984, 34(5): 975–986. doi: [10.1007/BF01009452](https://doi.org/10.1007/BF01009452).
- [16] HARDY G H, LITTLEWOOD J E, and PÓLYA G. Inequalities[M]. New York: Cambridge University Press, 1952: 261–262.
- 张少军: 男, 1989年生, 博士生, 研究方向为软件定义网络。  
兰巨龙: 男, 1962年生, 教授, 研究方向为宽带信息网、可重构柔性网络。  
江逸茗: 男, 1984年生, 助理研究员, 研究方向为软件定义网络、网络虚拟化。  
孙鹏浩: 男, 1992年生, 博士生, 研究方向为软件定义网络。