

# 云原生的通信大数据平台中 SDN 控制器部署方法

李景文<sup>†</sup>, 罗鹏

(中国移动通信集团广西有限公司, 广西 南宁 530022)

**摘 要:**研究了云原生的通信大数据平台中 SDN 控制器部署方法, 以提高 SDN 控制器部署的灵活性、高可用性, 更好地满足通信大数据平台需求。构建了云原生的通信大数据平台架构, 将原始通信大数据上传到 Kubernetes 集群中, 将其存储于 HDFS, 并导入至 Hive 数据资源仓库中, 利用 SDN 交换机完成通信数据转发。构建以最低通信代价为目标, 满足一台交换机仅能由一个控制器调度等约束的部署方案, 最终通过终端层展示处理结果。研究结论表明, 该部署方法不仅能够满足通信大数据平台的实时处理需求, 还能在保证系统稳定性的同时, 提供高效的数据处理和分析能力。通过实际应用验证, 该方法在优化平台通信代价、平均失效率、平台弹性以及通信时延方面表现出色, 为云原生通信大数据平台的构建和运维提供了有力的技术支持。

**关键词:**云原生; 通信大数据; SDN 控制器部署; Kubernetes 集群; 通信代价; 控制域

**中图分类号:** TP393

**文献标识码:** A

## SDN Controller Deployment Method in Cloud Native Communication Big Data Platform

LI Jingwen<sup>†</sup>, LUO Peng

(China Mobile Group Guangxi Co., Ltd., Nanning, Guangxi 530022, China)

**Abstract:** Research on the deployment method of SDN controllers in cloud native communication big data platforms to improve the flexibility and high availability of SDN controller deployment, and better meet the needs of communication big data platforms. Build a cloud native communication big data platform architecture, upload raw communication big data to Kubernetes clusters, store it in HDFS, and import it into Hive data resource warehouses, using SDN switches to complete communication data forwarding. Build a deployment plan with the goal of minimizing communication costs, satisfying the constraints that a switch can only be scheduled by one controller, and ultimately display the processing results through the terminal layer. The research conclusion shows that this deployment method can not only meet the real-time processing requirements of communication big data platforms, but also provide efficient data processing and analysis capabilities while ensuring system stability. Through practical application verification, this method has shown excellent performance in optimizing platform communication costs, average failure rates, platform resilience, and communication latency, providing strong technical support for the construction and operation of cloud native communication big data platforms.

**Key words:** cloud native; communication big data; SDN controller deployment; Kubernetes cluster; communication cost; control domain

信息技术的飞速发展和数字化转型的深入推进,使通信大数据平台成为支撑现代社会运行的重要基础设施之一<sup>[1]</sup>。软件定义网络(Software Defined Network, SDN)作为网络领域的一项重要技术<sup>[2,3]</sup>,其核心理念是通过软件编程的方式实现网络资源的灵活配置和管理,为通信大数据平台的网络架构带来了革命性的变化。SDN 控制器的部署是构建通信大数据平台的关键环节之一<sup>[4]</sup>。控制器作为 SDN 网络的核心组件,负责集中管理和控制网络流量,实现网络资源的动态分配和优化。然而,SDN 控制器的部署面临着诸多挑战<sup>[5]</sup>。

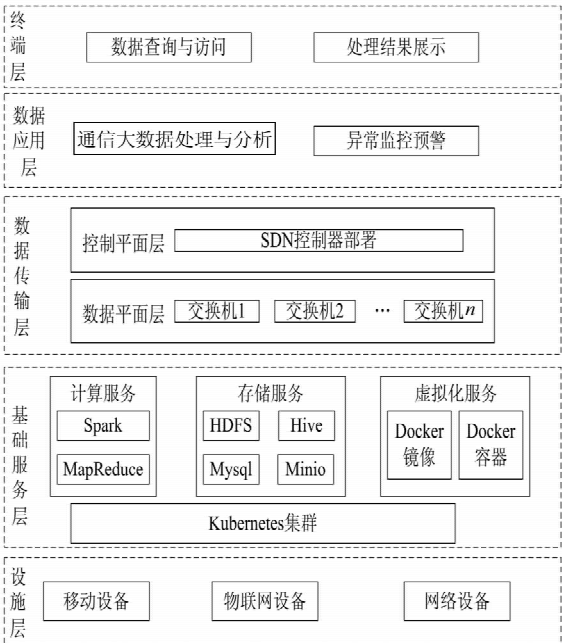
谢燕等设计以最低时延、成本为目标,同时满足负载均衡的 SDN 多控制器部署方案,通过遗传算法对其进行优化,确定最优部署方案。该方法更侧重于静态控制器的部署优化,当网络环境和负载情况发生显著变化时,该方法可能不再适用<sup>[6]</sup>。陈俊彦等采用 K-means++ 算法对网络区域进行聚类,得到多个控制域,基于最低成本、负载、流量均衡目标进行各分区控制器位置的优化选择。K-means++ 算法主要基于静态网络拓扑进行划分,无法适应网络设备增删改以及网络链路故障而发生动态变化情况<sup>[7]</sup>。曹绍华等为了解决不合理控制器部署对网络性能的影响,利用贪婪策略改进后的遗传算法对控制器部署问题进行优化,该方法在面对节点失效或链路中断等故障时,网络连通性、容错性较差<sup>[8]</sup>。Hussain 等根据吸引力原则筛选控制节点,将最小时延及控制链路可靠性作为控制器部署的优化目标,并利用改进人工鱼群算法对其该问题进行优化,该部署方案下的网络资源分配无法满足均衡性要求<sup>[9]</sup>。云原生强调将应用程序和服务设计为云环境下的原生应用<sup>[10]</sup>,通过容器化技术将 SDN 控制器及其相关组件打包成容器,可实现应用程序的可移植性和可扩展性。另外,在云原生环境下,SDN 控制器可以更容易地实现分布式部署和负载均衡,从而确保在单个节点出现故障时,其他节点能够迅速接管并继续提供服务。同时,云原生可以根据 SDN 控制器的实际需求动态分配和释放资源,从而实现更高效的资源利用。因此,本文提出云原生的通信大数据平台中 SDN 控制器部署方法,以提高 SDN 控制器部署的灵活性、高可用性等,更好地满足通信大数据平台需求。

1 SDN 控制器部署

1.1 云原生的通信大数据平台架构

云原生的通信大数据平台架构如图 1 所示,通信大数据的高效传输建立在设施层和基本服务层

之上,同时为数据应用层和终端层提供数据支持。设施层具有数据采集与汇聚功能,该层通过移动设备、物联网设备以及交换机设备等进行通信大数据的采集后,将其上传到基础服务层的 Kubernetes 集群中。基础服务层通过在各物理服务器上部署 Kubernetes 容器编排系统,构建适合通信大数据处理的容器云运行环境,以充分利用集群中物理服务器的计算资源和存储能力,同时能够根据应用需求自动调整容器的数量和资源分配。通过将 MapReduce 计算、HDFS 分布式存储等服务打包成一个独立的、可移植的容器镜像,并部署在 Kubernetes 集群中,实现计算资源的动态分配和扩展。数据传输层采用软件定义网络(SDN)进行通信大数据传输,该网络由数据平面层、控制平面层构成,数据平面层由 SDN 交换机等网络通用硬件组成,负责数据的转发和传输;控制平面层由 SDN 控制器实现,负责网络的集中化管理和数据平面行为的控制。该层在 SDN 控制器部署问题描述的基础上,通过计算节点相似度进行初始控制域分组,通过分裂处理实现分组结果的优化,在各控制域内构建以最低通信代价为目标函数,且满足一台交换机仅能由一个 SDN 控制器调度等约束的部署方案,实现控制器位置的确定。数据应用层通过北向接口与控制平面进行通信,提供网络服务和应用功能,可实现通信大数据处理与分析、网络设备性能、流量状况、安全事件等实时监控与预警等功能。终端层作为用户与平台直接交互界面,不仅可为用户提供数据查询与访问服务,也可实时接收应用层返回的分析结果和告警信息,并将其展示给用户。



、 图 1 云原生的通信大数据平台架构

1.2 基于 Kubernetes 的通信大数据存储管理

基础服务层通过构建数据资源仓库对通信大数据进行集中保存,其结构如图 2 所示。基于数据资源仓库的通信大数据存储过程如下:(1)将原始通信大数据传输给分布式文件系统(HDFS),编写一个配置文件用于指定连接参数、查询语句等,将其保存在一个方便访问的位置,在此基础上进行文件导入任务的构建,包含将 HDFS 中通信数据导入到 Hive 仓库所需的所有必要信息。(2)对导入任务进行类别划分后,将其分别置于不同管道中,根据排队机制对其进行管理。(3)利用处于闲置状态的不同类型解析器完成对应文件导入任务的加载处理后,在运行区对其进行读入、格式转换等处理,并保存到 Hive 数据仓库中。文件导入任务处理失败后,Kubernetes 可通过 Pod 的重启策略再次触发该任务,确保任务最终能够成功完成。用户无需关心通信数据的格式和来源,通过 Hive SQL 进行查询,即可获取其所需通信数据。利用各类型解析器将通信大数据由 HFDS 导入到 Hive 仓库,可实现通信大数据的集中化管理。

本文利用 Kubernetes 的容器化等特性,为通信大数据平台提供了更为高效、灵活和可靠的数据存储和查询服务<sup>[11]</sup>。Kubernetes 能够自动调度 HDFS 组件到集群合适节点上,确保数据的可靠性和高性能访问。同时,Kubernetes 的自动扩展机制可以根据集群的负载情况动态调整 HDFS 节点的数量,以满足不断变化的存储需求。Kubernetes 的资源管理和调度能力也可确保 Hive 任务能够得到足够的计算资源,从而实现快速的数据处理和查询。

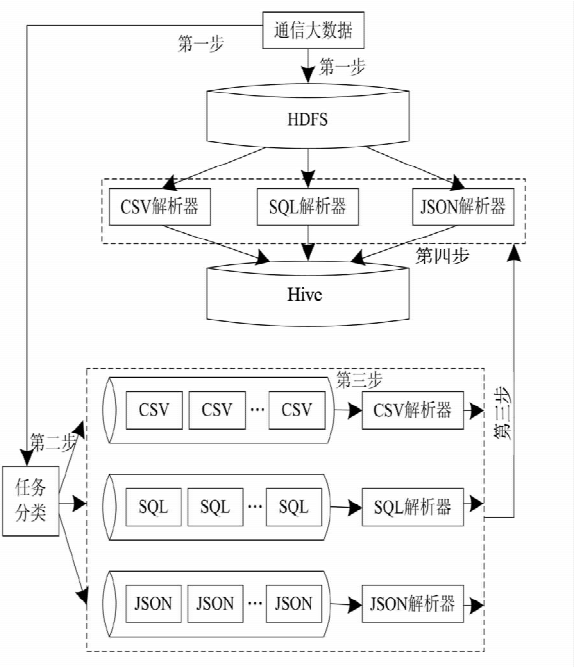


图 2 数据资源仓库基本结构图

1.3 通信大数据平台 SDN 控制器部署问题描述

通过无向图  $G(N,E)$  描述云原生的通信大数据平台 SDN 数据传输网络,网络中  $n$  个交换机  $n_i$  构成的集合表示为  $N = \{n_1, n_2, \dots, n_n\}$ ;数据传输链路集合表示为  $E = \{[(n_i, n_j) | n_i, n_j \in N]\}$ 。控制器所管辖控制域内的交换机集合表示为  $F \subseteq N$ 。控制器单位时间内处理的通信流数据总量表示为  $A = [a_1, a_2, \dots, a_{|F|}]$ ,SDN 控制器部署情况通过矩阵  $H = [h_{ij}]_{|F| \times |O|}$  描述,其中  $h_j$  可定义为二元组  $(\alpha_{ij}, \beta_{ij})$ ,对于第  $i$  个控制器,其在第  $j$  层所在位置表示为  $\alpha_{ij}$ ,其根节点所在位置表示为  $\beta_{ij}$ ,控制层的层级数表示为  $|O|$ 。云原生架构通信大数据平台 SDN 网络交换机部署情况通过  $X = [\chi_{im}]_{|N| \times |F|}$  表示,若  $\chi_{im} = 1$ ,说明第  $i$  个交换机已分配到第  $m$  个控制器控制域中;若  $\chi_{im} = 0$ ,则说明该交换机未分配。通信量矩阵通过  $T = [\tau_{ij}]_{|N| \times |N|}$  表示,单位时间内交换机节点之间数据通信流总量表示为  $\tau_{ij}$ 。交换机节点  $n_i$  的权重  $w(n_i)$  通过其传输的数据流数进行反映,计算公式描述为:

$$w(n_i) = \sum_{j \in E} \tau_{ij} \tag{1}$$

对于第  $i, j$  个交换机,二者之间最小长度传输链路表示为  $r_{ij}$ ,其权重  $w(r_{ij})$  通过该链路传输的所有数据流数之和进行反映,计算公式描述为:

$$w(r_{ij}) = \sum_{k \in N} \sum_{h \in N} \tau_{kh}, r_{ij} \subseteq r_{kh} \tag{2}$$

$L_{ij} = d_{ij}/b_{ij}$  用于描述数据流从交换机  $n_i$  经过最少跳数传输到交换机  $n_j$  需付出的通信开销,最短传输链路长度表示为  $d_{ij}$ ,最短传输链路带宽表示为  $b_{ij}$ 。

控制器负载定义为单位时间内控制器可调度的数据流总量,计算公式描述为:

$$\text{Load} = \sum_{i \in N} \sum_{j \in N} \tau_{ij} \chi_{im} \tag{3}$$

区域负载是所有控制器负载的集合,它反映了整个控制域中所有控制器处理流量的总体情况<sup>[12,13]</sup>,通过下式进行描述:

$$\text{Load} = \{\text{Load}_m | 1 \leq m \leq |F|\} \tag{4}$$

SDN 控制器部署效果可通过流表构建代价的高低进行反映<sup>[14]</sup>,流表构建通过三步实现,具体为:

(1)交换机提交流申请至 SDN 控制器,该步代价通过下式进行定义:

$$C_p^i = \sum_{i \in N} \sum_{m \in F} \tau_{ij} \chi_{im} d_{im} \tag{5}$$

其中:交换机至控制器  $m$  最短传输链路长度表示为  $d_m$ 。

(2)控制器接收流申请后,需计算出数据包应该经过的路径以及相应的处理规则,该步代价通过

$C_p^Q$  表示。

(3) 控制器将计算出流表路径和处理规则编码为流表项,并通过 OpenFlow 协议发送给交换机后,会将其本地安装。此步代价通过  $C_p^L$  表示,计算公式描述为:

$$C_p^L = \sum_{i \in N} \sum_{j \in N} \sum_{m \in F} \tau_{ij} \chi_{im} d_{im} \quad (6)$$

流表构建代价通过下式计算:

$$C_p = C_p^I + C_p^Q + C_p^L \quad (7)$$

#### 1.4 基于相似度的控制域划分

本文通过计算网络节点相似度实现通信大数据平台 SDN 通信网络控制域的初始分组,经过分裂处理实现控制域分组结果的优化。

##### 1.4.1 初始控制域分组

本文以域内交换机数量最大为目标,通过聚类进行通信大数据平台 SDN 网络控制域的初始分组,尽可能避免后续进一步对控制器数量进行处理。控制域初始分组步骤如下:

第一步:对通信大数据平台 SDN 通信网络中所有交换机节点之间的相似度进行运算;设定  $\lambda$  为最大循环次数,创建一个空队列 queue,用于存储待处理交换机节点;将 false 作为初始遍历标记,用于区分交换机节点是否已遍历。

第二步:在通信大数据平台 SDN 通信网络中选择一个边缘节点  $i$ ,以之为起始点,获得控制域  $D(i)$ 。

第三步:对  $D(i)$  相邻节点进行搜索,若交换机节点  $j$  与  $D(i)$  相似度最高,同时符合以下两个条件时,则将其放入新  $D(i)$  中,并将标签设置为 true。条件 1:新  $D(i)$  中交换机节点  $n$  至其他节点的通信时延满足最大时延  $\sigma$  约束;条件 2:新  $D(i)$  中交换机总数不超过  $\eta$ 。反复循环此过程,当  $D(i)$  不再有变化后停止。

第四步:将  $D(i)$  的邻居节点全都置于队列 queue 内。

第五步:当 queue 中不包含任何交换机节点后,转入第七步;反之,从队列中选取一个标签为 false 的交换机节点,当其与之邻域控制器的数据通信时延低于  $\delta$ ,则获得新控制域  $D(i)$ 。

第六步:对控制域  $D(i)$  的邻居节点  $j$  进行搜索,并检查其标签,当遍历到标签 false,则计算其与控制域  $D(i)$  的相似度,并与其他控制域的相似度进行比较,目的是找到相似度最大且可使以下三个特定条件同时成立的节点  $j$ 。条件 1:在节点  $j$  与  $D(i)$  组成的新控制域中,具有能使其与全部节点之间数据通信时延不高于  $\sigma$  的节点  $v$ ;条件 2:相邻两个控制域中局部控制器之间的通信时延不高于  $\delta$ ;条件 3:新  $D(i)$  交换机数量不超出  $\eta$ 。反复循

环此过程,当  $D(i)$  不再有变化后,转入第四步。

第七步:清空队列 queue,同时,所有交换机节点标志重置为 false,表示交换机节点尚未被分配到任何控制域中。

第八步:反复进行步骤二至步骤七,依据最低控制域数量、控制器容量利用率方差目标进行控制域初始分组。

第九步:停止迭代,获得控制域分组结果。

##### 1.4.2 控制域分组优化

控制域分组优化的目的是降低控制域数量,并寻求控制规模的均衡,以增强通信大数据平台稳定性和可靠性,避免出现资源利用不均衡问题<sup>[15]</sup>。本文通过分裂实现初始控制域分组结果的寻优,核心思想是对存在重合可能性较高的控制域进行处理,通过将全部交换机加入到相邻控制域中,实现控制域数量的有效控制。

控制域分裂实现过程为:

(1) 依据域内交换机数量由高到低顺序对分组后的控制域进行排列。

(2) 对各控制域  $D(i)$  进行搜索,并确定其邻居控制域。

(3) 分裂  $D(i)$ ,将其节点放置于邻居控制域  $D(k)$  中,当  $D(k)$  交换机总量超出最大数量  $\eta$  限制要求,则表明控制域分裂不成功,转入(5)。

(4) 对  $D(i)$  中节点分组,分组数量与其邻居控制域数量相同。

(5) 遍历各个组,检测将其加入到  $D(k)$  中是否能使  $\sigma, \delta, \eta$  条件同时满足,条件成立,则可实现控制域分裂,同时对控制域进行调整;反之,重复以上步骤。

(6) 判断  $D(i)$  是否全部遍历,已完成遍历,则转入停止分裂;反之,转入(2)。

#### 1.5 SDN 控制器部署

通过  $k = \sum_{n=1}^M \frac{R_n}{\alpha_c U_c}$  对通信大数据平台 SDN 控制器数量进行预测,通信大数据平台数据通信请求总量表示为  $\sum_{n=1}^M R_n$ ,SDN 控制器对通信请求的调度能力表示为  $\alpha_c U_c$ 。以通信代价最低为目标实现 SDN 控制器的部署,目标函数描述公式为:

$$\min(\text{Total}) \quad (8)$$

其中:通信大数据平台通信代价表示为 Total,用于描述交换机之间进行数据传输的开销,计算公式描述为:

$$\text{Total} = C_p + \sum_{i \in N} \sum_{j \in N} \tau_{ij} \times l_{ij} \quad (9)$$

通信大数据平台 SDN 控制器部署需遵循如下约束条件:

$$\forall i \in N : \sum_{m \in F} \chi_{im} = 1 \tag{10}$$

$$\chi_{im} \leq \sum_{m \in F} y_{m,j} \leq 1, \forall m \in F, \forall i \in N \tag{11}$$

$$\sum_{m \in F} y_{m,j} \leq 1, \forall j \in N \tag{12}$$

公式(10)约束交换机*i*仅能由控制器*m*调度；公式(11)表示部署在特定位置的控制器必须是工作的、可访问的，并且能够响应来自交换机的请求，且控制器*m*的位置具有唯一性。公式(12)表示*j*位置只能部署一个控制器。

交换机提交的流申请不能高于 SDN 控制器的最大容量占用率，公式描述为：

$$\sum_{i \in N} \chi_{im} R_i \leq \alpha_m U_m, \forall m \in F \tag{13}$$

2 实验分析

以某云原生通信大数据平台为实验对象，该平台网络拓扑中共有 34 个节点，平台服务器运行环境为采用 Linux 操作系统，CPU 选用具有强大计算能力和并行处理能力的 Intel Xeon 系列，采用 DDR4 ECC 高容量内存，以确保数据的快速处理和稳定性。网络带宽 10 Gbps，交换机全部采用华为 CloudEngine S16700 系列。应用研究方法对云原生通信大数据平台中的 SDN 控制器进行部署，通过对控制器部署结果以及平台性能等进行分析，验证研究方法的应用性能。

为分析研究方法的控制域分组能力，将基于 K-means 的 SDN 控制器部署方法、基于贪婪策略的 SDN 控制器部署方法(GCP)作为对比方法，选用控制器容量利用率(*R*)、控制域连通强度(*K*)作为评价指标对不同控制域分组结果进行评价，实验结果如表 1 所示。

表 1 不同方法下控制域分组结果对比分析

| 控制域<br>分组结果 | K-means  |          | GCP      |          | 研究方法     |          |
|-------------|----------|----------|----------|----------|----------|----------|
|             | <i>R</i> | <i>K</i> | <i>R</i> | <i>K</i> | <i>R</i> | <i>K</i> |
| D(1)        | 0.71     | 0.61     | 0.52     | 0.49     | 0.81     | 0.68     |
| D(2)        | 0.81     | 0.49     | 0.52     | 0.49     | 0.83     | 0.56     |
| D(3)        | 0.62     | 0.55     | 0.91     | 0.48     | 0.91     | 0.61     |
| D(4)        | 0.59     | 0.62     | 0.62     | 0.64     | 0.90     | 0.81     |
| D(5)        | 0.73     | 0.79     | 0.43     | 0.52     | —        | —        |
| D(6)        | —        | —        | 0.51     | 0.52     | —        | —        |

分析表 1 可知，应用研究方法对云原生通信大数据平台 SDN 网络进行控制域分组，获得的分组数量少于对比方法，*R* 指标达到 0.8 以上，*R* 指标定义为域内交换机数量与控制器最大容量之比，该

指标越大，控制域内交换机数量越多，控制器容量能够得到有效利用。当控制域数量达到 4 时，*K* 指标值上升到最大，达到 0.81。研究方法将节点相似度作为分组依据，使域内节点分布更加紧密，能够通过不同链路使交换机与控制器建立关系，极大增强了网络连通性的同时，使通信大数据平台更具弹性优势。GCP 方法将网络节点划分成 6 个控制域，边缘节点的存在对控制域的均衡性造成极大影响。K-means 方法将网络节点分成 5 个控制域，域内节点差异较大，降低了交换机与控制器之间的连通性，该方法以负载均衡目标进行控制域分组，故控制器容量利用率优于 GCP 方法。

在控制域分组的基础上，应用上述三种方法对 SDN 控制器进行部署，通过对不同方法下的平台通信代价、平均失效率、通信时延、平台弹性指标进行差异分析，验证研究方法在 SDN 控制器部署上的优越性，实验结果如表 2 所示。

表 2 不同方法下的 SDN 控制器部署性能对比

| 方法                   | 平台通信<br>代价        | 平均<br>失效率 | 平台<br>弹性 | 通信时<br>延/s |
|----------------------|-------------------|-----------|----------|------------|
| 基于 K-means 的<br>部署方法 | $4.5 \times 10^4$ | 0.27      | 0.65     | 1.24       |
| 研究方法                 | $3.8 \times 10^4$ | 0.22      | 0.71     | 1.1        |
| GCP 方法               | $4.9 \times 10^4$ | 0.32      | 0.59     | 1.35       |

分析表 2 可知，应用研究方法对云原生通信大数据平台中的 SDN 控制器进行部署，可使平台通信代价降至最低，仅为  $3.8 \times 10^4$ ，通信时延也最低。平均失效率指标反映通信大数据平台是否稳定，二者具有反比例关系，经研究方法对 SDN 控制器进行部署后，平均失效率指标为 0.22，低于对比方法，云原生通信大数据平台稳定性更好。平台弹性指标反映了网络在面临故障时的恢复能力。研究方法的弹性值达到 0.71，高于对比方法，意味着研究方法在处理节点失效或链路中断等故障时具有更强的连通性和容错能力。实验结果表明，研究方法在降低云原生通信大数据平台通信代价、通信时延的同时，提高了平台稳定性、连通性以及容错性。

通过对 SDN 控制器部署结果进行分析，验证研究方法的部署性能，实验结果如图 3 所示。

分析图 3 可知，研究方法将云原生的通信大数据平台中的 34 个网络节点分成四个控制域，各控制域中交换机数量基本一致，控制域规模处于相对平衡状态，4 个控制器节点均通过红色矩形框进行标记，其编号分别为 5、20、27、34，交换机节点可通过多条链路与控制器的建立联系，既可保证网络连通性，又

可使云原生的通信大数据平台具有更高弹性。

设定云原生的通信大数据平台运行环境中  
有不同大小数据包传输任务,通过对比基于 K-means  
的 SDN 控制器部署方法、基于贪婪策略的 SDN 控  
制器部署方法以及研究方法部署下的平台功耗差  
异,分析研究方法的控制器部署能力,实验结果如  
图 4 所示。

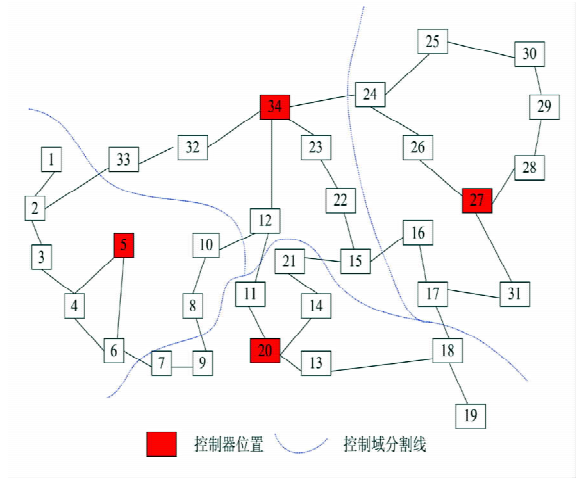


图 3 SDN 控制器部署结果分析

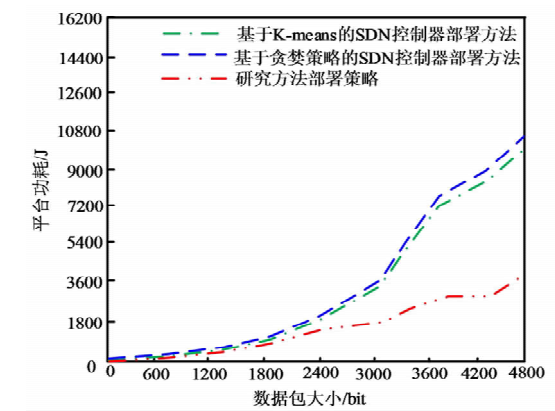


图 4 不同控制器部署策略对平台功耗的影响

分析图 4 可知,随着数据包的不  
断增大,平台数据传输功耗成逐渐增长规律变化,当数据包小于 1800 bit 时,三种部署策略下的平台功耗波动均极小,且功耗曲线差异不明显;平台在传输更大数据包时,不同部署策略下的平台功耗差异开始逐渐变大,当数据包达到 4800 bit 时,基于 K-means 的 SDN 控制器部署方法、基于贪婪策略的 SDN 控制器部署方法部署下的平台功耗达到 10140 J,通信大数据平台长期在该状态下运行,无疑会带来巨大的能源开销;采用研究方法下的控制器部署策略,平台功耗仅为 3730 J,仅为传统控制器部署下的一半左右,显著的功耗差异体现了研究方法的有效性。

3 结论

以某云原生通信大数据平台为实验对象,通过对不同方法下的控制域分组结果、控制器部署结果以及数据包大小对平台功耗影响验证研究方法的部署效果,实验结果表明:

(1)研究方法将网络节点划分成 4 个控制域,并完成控制器位置的确定,R 指标、K 指标值均高于对比方法。

(2)随着数据包的不  
断增大,研究方法依然可将通信大数据平台功耗控制在 3730 J,具有显著优越性。

参考文献

[1] 王芳鸣,王斌,陈志浩,等. 基于 SDN 的匿名通信系统设计[J]. 计算机工程与设计,2021,42(7):1831—1836.

[2] 朱宇昂,赵亚丽,赫佳峦,等. 基于分布式 SDN 的移动通信系统拓扑发现方法[J]. 系统工程与电子技术,2024,46(1):357—365.

[3] 高荣亮,郑乐. 高可用 SDN 控制器主从分布式系统设计[J]. 光通信技术,2021,45(7):59—62.

[4] 尚秋峰,李炎,黄尔杰. 电力通信网中 SDN 控制器部署方法[J]. 电测与仪表,2020,57(16):78—82.

[5] 陈金涛,梁俊,刘波,等. 5G-卫星集成网络中控制器与网关可靠部署策略[J]. 空军工程大学学报(自然科学版),2021,22(3):68—73.

[6] 谢燕,张爽. SDN 中多控制器均衡部署优化方法仿真[J]. 计算机仿真,2023,40(7):378—381+392.

[7] 陈俊彦,李玥,梁楚欣,等. SDN 多控制器部署及流量均衡研究[J]. 计算机工程与科学,2021,43(5):830—835.

[8] 曹绍华,付鲁非,陈舒,等. 基于改进遗传算法的 SDN 控制器放置优化研究[J]. 中国电子科学研究院学报,2023,18(1):26—34.

[9] HUSSAIN M W, KHAN M S, REDDY K H K, et al. Extended indirect controller-legacy switch forwarding for link discovery in hybrid multi-controller SDN[J]. Computer communications, 2022,189(May):148—157.

[10] 王占林,辛亚芳,马亮,等. 基于云原生的时空大数据平台设计与实现[J]. 地理信息世界,2022,29(2):94—98.

[11] 徐小华,胡忠旭,常凤. 基于云原生的数智校园基础平台系统的架构设计与实现[J]. 现代计算机,2022,28(21):117—120.

[12] 张元龙,廖晓群,张佳庚. 基于网络拓扑划分的大规模 SDN 控制器部署[J]. 计算机工程与设计,2022,43(9):2485—2492.

[13] 黄梅根,袁雪,吴令令,等. 软件定义网络中基于时延和负载的多控制器部署策略研究[J]. 电子与信息学报,2022,44(1):288—294.

[14] 马勤,李志芳. SDN 中一种多控制器负载均衡方案[J]. 西南师范大学学报(自然科学版),2021,46(7):114—119.

[15] 郭子桢,梁俊,肖楠,等. 软件定义卫星网络多控制器可靠部署算法[J]. 西安交通大学学报,2021,55(2):158—165.