文章编号:1671-4598(2023)01-0257-07 DOI:10.16526/j.cnki.11—4762/tp.2023.01.038 中图分类号:TP39 文献标识码:A

基于 SDN 网络大象流负载均衡算法研究

张一凡，韩卫占
（中国电子科技集团公司 第五十四研究所，石家庄 050081）

摘要：负载均衡算法是通过对网络中的流量进行调度来提高网络资源利用率，是计算机网络中的一个重要研究方向；针对网络中大象流导致的网络拥塞和老鼠流的排队时延等负载不均衡问题，提出了带宽和时延加权负载均衡（BD-WLB）算法来提高负载均衡性能，综合考虑了大小流之间的流量特征不同，改进了传统算法的路径计算方式；算法通过控制器来获取网络流量和状态信息；然后利用带宽和时延等网络状态参数来为大象流和老鼠流分别计算最优路径；采用 P4 语言来对数据平面转发流程进行优化处理；实验结果表明，在高负载状态时，BD-WLB 算法相比于 ECMP 算法提高了 38.4%的网络吞吐量和 41.9%的链路利用率，降低了 41.8%的网络时延；使网络资源得到了更好的利用，证明了 BD-WLB 算法的可行性和有效性。

关键词：计算机网络；软件定义网络；负载均衡；协议无关的包处理器；大象流检测

Research on Elephant Flow Load Balancing Algorithm Based on SDN

ZHANG Yifan, HAN Weizhan
(China Electronics Technology Group Corporation No.54 Research Institute, Shijiazhuang 050081, China)

Abstract: Load balancing algorithm is an important research direction in computer network, which can improve the utilization of network resources by scheduling network flow. Aimed at the load imbalance problems of network congestion caused by elephant flow and queue delay of rat flow in network, a bandwidth and delay weighted load balancing (BD-WLB) algorithm is proposed to improve the load balancing performance. The path calculation method of the traditional algorithm is improved by considering the different traffic characteristics between small and large streams. The algorithm obtains network traffic and state information through the controller. Then the network state parameters of bandwidth and delay are used to calculate the optimal path for elephant flow and mouse flow respectively. P4 language is used to optimize the data plane forwarding process. The experimental results show that compared with the ECMP algorithm at high load, the BD-WLB algorithm improves the network throughput by 38.4%, the link utilization by 41.9% and the network delay by 41.8%. It makes better use of network resources, and proves the feasibility and effectiveness of the BD-WLB algorithm.

Keywords: computer network; SDN; load balancing; P4; elephant flow detection

0 引言

随着互联网技术的迅速发展，近年来大数据[1]、云计算[2]、5G[3]等新兴技术也在飞速步入应用阶段。互联网在人们日常生活中占据的比重越来越大。网络用户与网络业务的规模也在迅速扩大，这就导致了网络的维护与管理难度的增加。为了提高网络服务质量，负载均衡技术[4]就成为了当前的研究热点。

负载均衡技术是指根据网络状况以及任务需求，将负载分摊到多个操作单元上运行，从而利用有限的资源来完成更多的任务。利用负载均衡技术主要避免了网络资源的不合理利用等问题，从而最大化地利用网络现有资源，是提高网络性能最常用的策略之一[5]。

负载均衡技术总体可分为软/硬件负载均衡、本地/全局负载均衡四大类[6]。其中应用最多的是软件负载均衡。通过部署软件来完成负载均衡，因此成本低廉，对于硬件设备的要求也很低。硬件负载均衡是通过部署负载均衡器，利用专用设备来完成负载均衡，因此成本要更高。但由于网络的规模巨大，很难统一得增加专门的硬件设备，因此硬件负载均衡仅在小规模网络中使用。本地负载均衡仅针对本地范围内的网络进行负载均衡，需要新增一台服务器来完成工作。全局负载均衡针对的是大规模网络场景，实现了地理位置无关性。但需要大量的软硬件设备来进行支持，成本极高。

在众多负载均衡算法中等价多路径路由（ECMP，equal cost multi-path）算法[7]是最常用的负载均衡算法。算法通过 CRC16 算法来计算数据流的五元组哈希值。然后将哈希值与转发路径条数进行取模运算，根据余数选择对应的转发路径，使同一条数据流的传输路径不变。理论上来说哈希值取模后的余数分布概率均等，因此可以一定程度上达到负载均衡的效果。但 ECMP 算法是静态算法，将所有路径都认为是等价路径，不考虑网络实际状态。因此无

收稿日期:2022-11-12； **修回日期**:2022-11-17。
作者简介:张一凡(1997-)，男，河北石家庄人，硕士研究生，主要从事计算机网络方向的研究。
通讯作者:韩卫占(1963-)，男，河北石家庄人，博士研究生，研究员级高级工程师，主要从事通信与信息系统方向的研究。
引用格式:张一凡，韩卫占.基于 SDN 网络大象流负载均衡算法研究[J].计算机测量与控制，2023，31(1)：257-263.

法达到最佳负载均衡效果。为了解决 ECMP 算法的弊端，文献［8］提出了 Hedera 算法。算法通过控制器来周期进行链路信息获取和大象流检测。针对大象流选择最先发现的能满足大象流转发带宽的路径进行转发，老鼠流选择基于哈希的 ECMP 算法进行转发。基于贪婪思想的 Hedera 算法虽然考虑了大象流的需求，但计算出的转发路径并不一定是最优路径。同时也没有考虑到老鼠流跟在大象流后的排队时延问题。但由于 Hedera 算法是一种动态负载均衡算法，会根据网络状态及时调整转发路径，因此负载均衡效果仍比 ECMP 算法优秀。

相比于静态负载均衡算法，软件定义网络（SDN，software defined network）的负载均衡通过控制器来获取全网状态信息，能够满足现代网络对高速数据传输[9]的需求。例如文献［10］提出的动态负载均衡算法，通过对网络状态的实时监测来动态调整流量，从而适应网络的变化。文献［11］通过控制器来分析网络节点的负载状况，然后通过调整源节点的发送速率来有效缓解网络负载状况。文献［12］针对大象流进行路径规划，以避免大象流之间产生冲突，减少网络拥塞的出现。以上算法的重点都是针对网络状态来实时调整负载，比传统网络的负载均衡算法更加优秀。但在数据流量较多的环境下，单个控制器可能无法完成所有功能，因此文献［13－15］通过部署多个控制器来减轻控制器负担，完成负载均衡等功能，来提高网络性能。

同时协议无关的可编程包处理器[16-17]（P4，programming protocol-independent packet processors）的出现实现了数据平面的可编程性。使得研究人员可以完整的定义数据包处理流程，也为负载均衡的发展提供了新的选择。文献［18］提出了一种基于可编程数据平面的负载均衡算法 HULA。通过探针实现对网络状态的实时感知，并根据感知信息来实时调整转发路径，提高了算法的可扩展性。文献［19］提出了 SilkRoad 算法，通过减少匹配与和动作的数量，可以使用一台可编程交换机来完成大量负载均衡工作，有效降低了成本。文献［20］提出的 Beamer 算法可以利用后端服务器来保存数据流的状态信息，减少了其他设备的工作量，提高了负载均衡性能。此外还可以借助 P4 语言来定制新的协议，文献［21］提出新协议 NDP，同时实现了大流的高吞吐量和小流的低时延传输。

为了解决大象流导致的网络拥塞[22]和排队时延[23]等负载不均衡问题，本文提出的 BD-WLB 负载均衡系统通过控制器来获取网络状态信息，同时进行大象流检测。数据平面针对大象流选择剩余带宽最大的路径转发，老鼠流选择路径时延最小的路径转发。并选择 ECMP 算法以及 Hedera 算法在网络吞吐量、链路利用率、网络时延三方面进行了比较分析。

1 BD-WLB 负载均衡系统架构

根据功能不同，可以将 BD-WLB 负载均衡系统划分为三个模块。分别为信息获取模块、负载均衡模块、数据平面转发模块三个部分。其具体架构如图 1 所示。

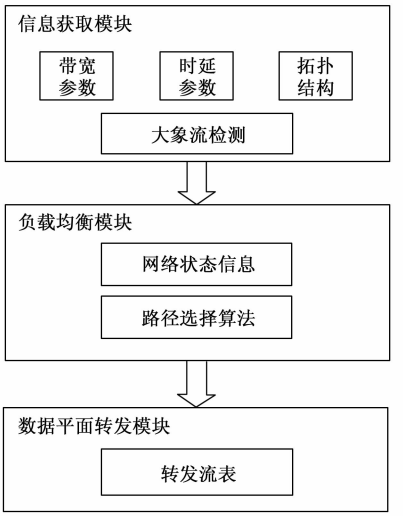

图 1 负载均衡系统整体架构

SDN 网络控制器拥有整个网络的网络资源视图，因此信息获取模块通过 SDN 网络控制器来获取网络拓扑结构以及各个交换机的流量数据。通过对流量数据的分析，可以获得网络链路的剩余带宽和链路时延等信息。同时还可以根据预先设定好的阈值，来判断该数据流是否为大象流。同时控制器还需对当前网络状态进行周期性检测，根据网络状况及时调整转发路径，避免网络拥塞的出现。负载均衡模块的工作是根据信息获取模块获得的网络状态信息对转发路径进行加权，针对大象流和老鼠流选择不同的加权公式计算最优转发路径。并将转发路径信息保存到转发表中，用于后续数据流的转发。数据平面转发模块的工作是根据转发表完成数据流的转发，使用 P4 语言优化转发处理流程。

2 BD-WLB 负载均衡系统实现

2.1 大象流检测

文献［24］研究发现，网络中的数据流量也存在着二八原则。百分之二十的大象流包含着网络中百分之八十的流量，百分之八十的老鼠流仅占网络流量的百分之二十。其中大象流对于网络带宽情况敏感，老鼠流对于网络时延情况敏感。因此为了避免上述问题导致的网络资源浪费，必须要将大象流和老鼠流区分开。为大象流选择剩余带宽最大的路径转发，老鼠流选择链路时延最短的路径转发，来确保网络资源得到充分的利用。

因此信息获取模块将通过检测交换机端口信息来确定数据流是否为大象流，其中大象流阈值设置为链路带宽的 10%，只要数据流的平均传输速率达到链路带宽的 10%，就认为该数据流为大象流。具体伪代码如图 2 所示。

2.2 BD-WLB 算法

为了尽可能减少网络拥塞和排队时延。本文提出的 BD-WLB 算法利用控制器获取网络实时状态信息，同时进行大象流检测。使用剩余带宽最大的路径转发大象流，时延最小的路径转发老鼠流。其中大象流最优路径的权值计算公

```
flow.length,flow.duration:
//输入数据流长度和数据流持续时间
flow.speed=(flow.length/flow.duration)
//计算数据流传输速度
link.occupancy.rate=flow.speed/bandwith
//计算数据流的链路占用率
if link.occupancy.rate>0.1
//如果占用率超过百分之十
elephant_flow=true
//确定为大象流
```

图 2　大象流检测伪代码

式如下所示。

式（1）用来计算大象流当前路径 j 的权值 E_j：

$$E_j = \left[\frac{\xi_j \cdot B_j}{\sum_{k=1}^{n}\xi_k B_k}\right]\times 100\% \tag{1}$$

其中：B_j 表示路径 j 的剩余带宽，$\sum_{k=1}^{n}\xi_k B_k$ 表示所有路径的剩余带宽和。

式（2）用来计算老鼠流当前路径 j 的权值 M_j：

$$M_j = \left[\left(\xi_j - \frac{\xi_j \cdot D_j}{\sum_{k=1}^{n}\xi_k D_k}\right)\right]\times 100\% \tag{2}$$

其中：D_j 表示路径 j 的时延，$\sum_{k=1}^{n}\xi_k D_k$ 表示所有路径的时延和。ξ_j 用于判断当前路径 j 是否为拥塞路径，算法中将链路剩余带宽小于 10% 的路径认为是拥塞路径。使用 Ba_j 来表示当前路径 j 的总带宽，那么 ξ_j 的表达公式如式（3）所示：

$$\xi_j = \begin{cases} 1, & B_j \geqslant 10\% Ba_j \\ 0, & B_j < 10\% Ba_j \end{cases} \tag{3}$$

从当前路径的带宽与时延权值信息中选择所有路径中具有最大权值信息的路径，将该路径作为最优转发路径，并保存到转发表中方便后续数据流转发。式（4）用来计算大象流所有权值路径中具有最大权值的路径：

$$W_{ej} = MAXE_j \tag{4}$$

式（5）用来计算老鼠流的最大权值路径：

$$W_{mj} = MAXM_j \tag{5}$$

BD-WLB 负载均衡系统转发数据流的具体流程如图 3 所示：

1）当第一个数据流到达交换机时，首先通过哈希算法计算数据流五元组哈希值得到流 ID。同时利用 SDN 网络控制器进行大象流检测。交换机根据获得的链路带宽与时延信息来计算大象流与老鼠流的最优转发路径并保存。其中大象流选择剩余带宽最大的路径转发，老鼠流选择时延最小的非拥塞路径转发。

2）后续数据流到达时，首先计算出数据流的流 ID 值，解析数据流的 IPv4 地址。然后使用流 ID 和目的 IP 地址对转发表进行关键字段匹配。匹配成功后根据转发信息转发该数据流。若数据流 ID 匹配但目的 IP 地址不匹配，则说明发生了哈希冲突，此时将流 ID 与目的 IP 地址相异或作为新的流 ID，然后利用 BD-WLB 算法计算最优转发路径，并保存到转发表中。

3）由于大象流的持续时间一般很长。为了避免大象流长时间使用同一路径转发，导致区域路径拥塞。同时最优路径随着网络状态的变化也在不断变化。因此针对转发表项设置了存在时间，从而过一段时间就重新计算大象流最优路径。同时存在时间也应大于老鼠流传输时间，从而避免老鼠流多次计算转发路径，占用计算资源。当数据流在转发表中无匹配项时，重复步骤（1），重新计算数据流的最优转发路径并保存，从而完成数据流转发。

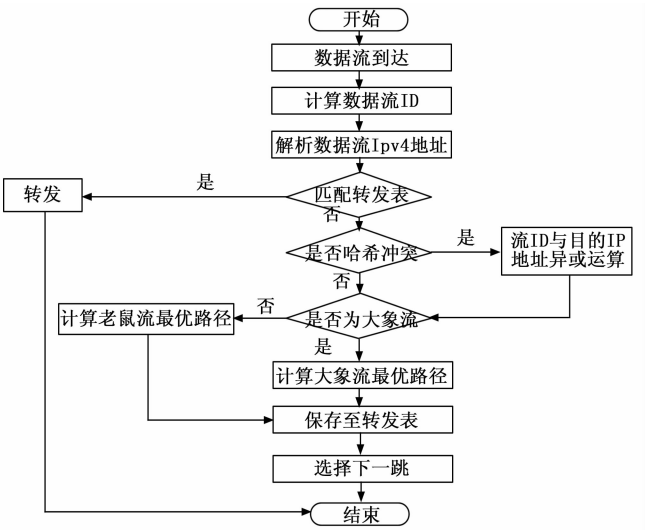

图 3　数据流转发流程图

2.3　P4 实现数据流转发

利用 P4 语言来实现数据平面数据流的整个处理过程，实现数据流的定制转发，使数据包处理过程更加灵活。一个完整的 P4 程序主要包含首部（Headers）、解析器（Parsers）、表（Tables）、动作（Actions）、控制程序（Control Programs）五个部分。其中路径选择动作表的工作是首先读取数据流的流 ID 和数据包的目的 IP 地址，进行最长前缀匹配，若能成功匹配，则根据匹配项转发数据包。若不能匹配，则判断是否出现哈希冲突。若出现则执行流 ID 与目的 IP 地址的异或运算。然后进行 BD-WLB 路径选择等操作。而执行路径选择动作时，根据当前的流 ID 值和相应的路径权值计算公式来进行路径选择。其中 BD-WLB 路径选择的伪代码如图 4 所示。

判断转发链路优劣的标准是根据式（1）～（5）来计算出的当前路径的状态参数信息，针对大象流选择具有最大链路带宽权值参数的路径作为最优转发路径。针对老鼠流选择具有最大时延参数的路径。并将选择的最优路径信息保存到转发表中，用于后续数据流的转发。其路径计算过程伪代码如图 5 所示。

流控制程序是将前面四个组件所定义的包头、解析流程、表、动作整合起来。用来定义数据流的处理和转发逻辑。流控制程序的伪代码如图 6 所示。

```
table bd_wlb_group{
    reads{
        flowlet_id,ipv4.dst_Addr:lpm;
          //最长前缀匹配
    }
    actions{
        flowlet_idXOR ipv4.dst_Addr;
        //计算新的流ID
        bd_wlb_select;
        //进行BD-WLB路径选择
    }
}
```

图 4 BD-WLB 路径选择伪代码

```
table W_link{
    reads{
        link.state():link_b.link_d;
//当前路径链路带宽、时延信息。
    }
    actions{
        max_Ej_link;
        //大象流最大权值路径
        max_Mj_link;
        //老鼠流最大权值路径
        drop;
    }
}
actions max_Ej_link(link_b){
//大象流选择具有最大路径带宽权值的链路
  modify_field(ingress_metadata_metadata.link_b);
  add_to_field(ingress_metadata_metadata.N_hop,1);
      //保存到转发表中
}
actions max_Mj_link(link_d){
//老鼠流选择具有最大路径时延权值的链路
  modify_field(ingress_metadata_metadata.link_d);
  add_to_field(ingress_metadata_metadata.N_hop,1);
      //保存到转发表中
}
```

图 5 BD-WLB 链路计算伪代码

3 性能评估

3.1 模拟环境搭建和参数设置

实验采用 ONOS 控制器和 Mininet 软件搭建实验仿真平台。网络拓扑结构如图 7 所示，使用 $k=4$ 的 Fat-tree 拓扑，包括 16 台主机、8 个接入层交换机、8 个汇聚层交换机、4 个核心层交换机。链路带宽设置为 100 Mbit/s，链路延时设置为 1 μs，并使用 iperf 工具产生模拟流量。

选择网络吞吐量、链路利用率、网络时延三个指标作为负载均衡性能指标。

其中网络的吞吐量是指在单位时间内通过网络传输并且能够成功接收到的数据总量。网络的吞吐量越高，说明性能越优。

链路利用率指的是数据传输中使用的链路数量与所有链路数量的比值。链路利用率可以反映出网络中传输链路的使用情况，若链路利用率越高，则说明网络流量分布均衡，网络资源利用越充分。反之则说明网络资源没有被充分利用。

```
control ingress{
  apply(flowlet);
  if(bd_wlb_hash=flowlet_bd_wlb_hash,
  ipv4.dst_Addr≠flowlet_ipv4.dst_Addr){
      //出现哈希冲突时
  apply(new_flowlet);
  //流ID与目的IP地址进行异或运算
      }
    apply(W_link);
    //执行最大权值计算
    apply(bd_wlb_group);
    //执行BD-WLB算法进行路径选择
    apply(bd_wlb_nhop);
    //执行下一跳
    apply(forward);
    //执行数据流转发
}
```

图 6 BD-WLB 流控制程序伪代码

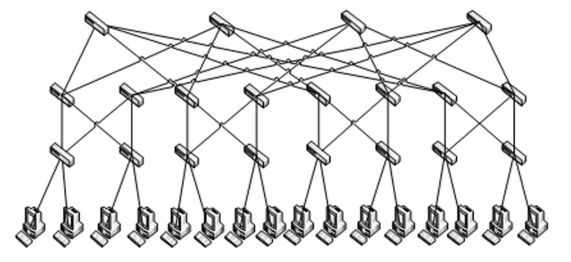

图 7 网络拓扑结构

网络时延指的是从源节点到目的节点的传输时间，传输时间越小则证明传输性能越好，网络服务质量越高。反之则说明网络可能出现了拥塞情况。

3.2 对比算法和流量模型

选择 ECMP 与 Hedera 两个经典负载均衡算法与本文提出的 BD-WLB 算法进行对比实验，来验证 BD-WLB 算法的可行性与有效性。

实验中通过 iperf 工具产生模拟数据流量。且大象流带宽范围为 10 Mbit/s 到 100 Mbit/s 之间，老鼠流带宽小于 10 Mbit/s。并且采用两种不同的数据流模型进行测试。第一种流量模型为 Staggered（pEdge，pPod）模型。该流量模型中主机会以概率 pEdge 向同一交换机下的主机发送流量，以概率 pPod 向同一个 Pod 内的主机发送流量，以概率 1－（pEdge＋pPod）向其余 Pod 内主机发送流量。同时该模型中大象流和老鼠流的比例按照实际网络中二比八的比例设置。实验中具体选择了 stag（0，0.1）、stag（0.1，0.1）、stag（0.1，0.2）、stag（0.2，0.2）、stag（0.2，0.3）、stag（0.3，0.3）、stag（0.4，0.3）、stag（0.6，0.2）、stag（0.8，0.1）九种流量模型。

第二种模型为 load _ x 模型，其中 x 代表大象流比例。通过调整大象流概率来改变网络负载。实验中具体选择了 load _ 0.1 到 load _ 0.9 共九种模型来验证算法性能。

3.3　实验结果与分析

如图 8 所示，在 stag 模型下的 9 种情况中，随着流量模型参数的变化，大部分流量从 Pod 间流量转为 Pod 内流量。此时三种算法的吞吐量都有提升。在前五种流量模型时，流量多数为 Pod 间流量，此时 BD-WLB 算法吞吐量要高于 ECMP 和 Hedera 算法，原因是 BD-WLB 算法综合考虑网络状态，分别为大象流和老鼠流选择满足其需求的最优路径。而 Hedera 算法的贪婪思想导致算法虽然考虑了大象流的带宽需求，但选择的路径很可能不是最优路径。ECMP 算法由于产生的大流碰撞，导致吞吐量最低。在 stag（0.3，0.3）、stag（0.4，0.3）、stag（0.6，0.2）、stag（0.8，0.1）四种情况时，流量多数为同一 Pod 内的流量。此时大流碰撞的情况减少，因此 ECMP 算法的吞吐量增加。Hedera 算法将大流转发路径进行了优化，因此吞吐量比 ECMP 高。此时 BD-WLB 算法的吞吐量虽然比 ECMP 和 Hedera 高，但三种算法的差距在缩小。因此 BD-WLB 算法对于 Pod 间流量的处理性能更优。

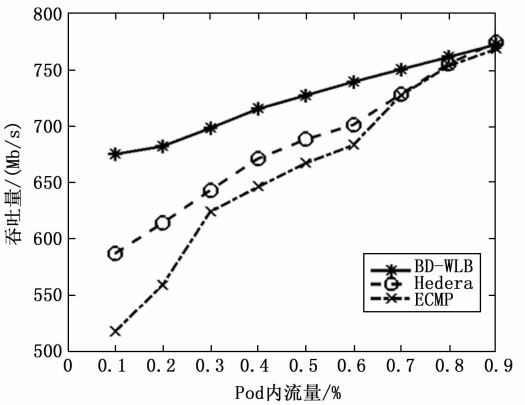

图 8　stag 模型网络吞吐量

如图 9 所示，在 load _ x 模型下随着流量负载的增加，三种算法的吞吐量都在增加。低负载状态时，三种算法的吞吐量差别不大。但在高负载状态时，BD-WLB 算法的吞吐量最大，相比于 Hedera 算法提升了 16.1%的吞吐量，相比于 ECMP 算法提升了 38.4%的吞吐量。这是由于 ECMP 算法易产生大流碰撞，导致网络拥塞，吞吐量下降。Hedera 算法依靠贪婪思想对大流进行路径优化，对小流采用 ECMP 算法进行转发，降低了大流碰撞发生的概率，因此在高负载状态时 Hedera 算法比 ECMP 算法的吞吐量高。BD-WLB 算法吞吐量最高是因为算法综合考虑网络状态，结合链路的带宽、时延参数对转发路径进行优化，在高负载状态下，能够为大象流和老鼠流找到最优转发路径，减少拥塞而实现更高的吞吐量。

如图 10 所示，在 stag 模型中的九种情况下，随着 Pod 间流量的减少，三种算法的链路利用率都在不断下降。这

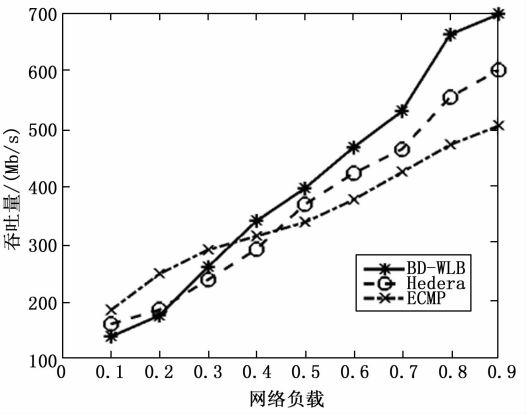

图 9　load _ x 模型网络吞吐量

是因为 Pod 间流量占大多数时，三种算法对数据流的转发都涉及到了上层的路径。并且 BD-WLB 算法通过实时监测链路的可用带宽和时延，并进行大象流检测，可以根据大象流和老鼠流的流量特征与需求来选择最合适的路径，因此链路资源利用更加充分，链路利用率最高。Hedera 算法的链路利用率比 ECMP 算法高，这是因为 Hedera 算法也会在检测到大流后，为大流选择第一条满足要求的路径，同样提高了链路利用率。而 ECMP 算法不考虑网络状态，数据流选定路径后，后续转发路径不变。无法利用网络中的冗余链路，因此链路利用率最低。

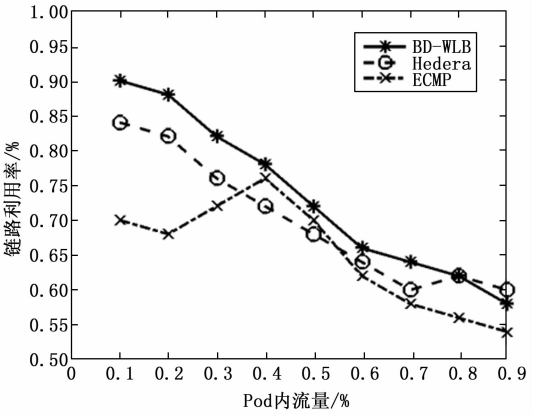

图 10　stag 模型链路利用率

如图 11 所示，随着流量负载的增加，BD-WLB 算法的链路利用率最高，其次是 Hedera 算法，ECMP 算法的链路利用率最低。这是由于 ECMP 在确定路径之后，后续的数据流转发路径一般不变，因此无法利用网络中的冗余链路。并且随着负载的增加，ECMP 不随网络状态更改路径，更易导致大流碰撞，因此 ECMP 算法的链路利用率低于另外两种算法，并且在高负载状态时链路利用率出现下降。Hedera 算法通过周期性的网络检测，针对大象流进行转发路径优化，针对老鼠流采用 ECMP 算法进行转发，从而可以利用网络中的冗余链路。因此 Hedera 算法的链路利用率要高于 ECMP 算法。相比另两种算法，BD-WLB 算法根据链路带宽和时延考虑了针对大小流的最优转发路径，尽可

能利用网络中的冗余路径。因此在高负载状态时 BD-WLB 算法的链路利用率要高于 ECMP 算法 41.9%，高于 Hedera 算法 7.3%。

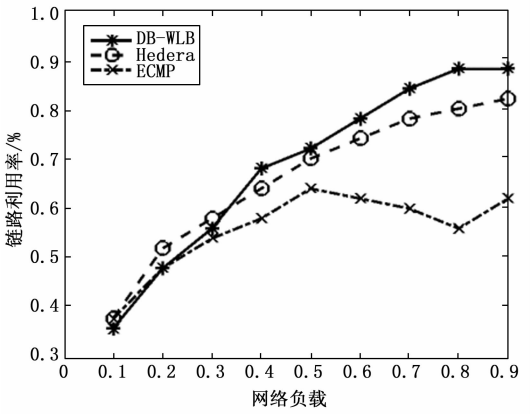

图 11 load _ x 模型链路利用率

如图 12 所示，在 stag 模型中，随着不同 Pod 间传输的流量比重降低，三种算法的传输时延均呈现下降趋势。在 stag（0，0.1）、stag（0.1，0.1）、stag（0.1，0.2）、stag（0.2，0.2）、stag（0.2，0.3）五种情况时，Pod 间传输的流量多，此时 BD-WLB 算法的传输时延最低，这是由于 BD-WLB 算法综合考虑了带宽和时延条件，为大象流和老鼠流选择了最优的转发路径。避免了大流碰撞导致的时延增加问题和老鼠流在大象流后的排队时延问题。在 stag（0.3，0.3）、stag（0.4，0.3）、stag（0.6，0.2）、stag（0.8，0.1）四种情况时，流量主要在同一 Pod 内转发，大部分流量无需算法进行调度，可以直接转发，因此三种算法的传输时延均在下降，并且时延相差不大。

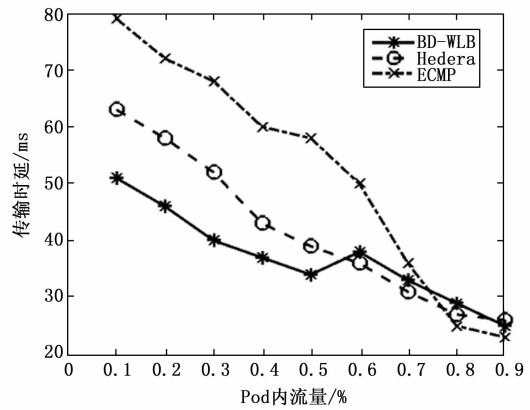

图 12 stag 模型传输时延

如图 13 所示为不同流量负载下 ECMP、Hedera 和 BD-WLB 算法的传输时延。在高负载状态时，BD-WLB 算法的传输时延最低。此时 BD-WLB 算法相比于 ECMP 算法降低了 41.8%的传输时延，相比于 Hedera 算法降低了 25%的传输时延。说明 BD-WLB 算法可以在高负载状态时达到很好地负载均衡效果，避免了链路拥塞和排队时延的出现。而 Hedera 算法虽然在一定程度上优化了大流的转发路径，但并未考虑对时延敏感的小流。没有解决小流跟在大流后的排队时延问题。因此 Hedera 算法的传输时延要高于 BD-WLB 算法。ECMP 算法对网络状态和流量大小均不关心，因此随着负载的增加，链路拥塞与排队时延等问题不断出现，因此在高负载状态下 ECMP 算法的传输时延最高。

当网络处于低负载状态时，BD-WLB 算法的传输时延要略高于 ECMP 算法和 Hedera 算法，这是因为 BD-WLB 算法需要进行网络链路信息的搜集、大象流检测、最优链路计算等过程，需要一定的时间。因此在低负载状态时 BD-WLB 算法的传输时延较高。由于低负载状态时小流占比例大，此时 Hedera 算法和 ECMP 算法都很少发生大流碰撞而提高时延。因此两种算法的传输时延十分接近且小于 BD-WLB 算法。

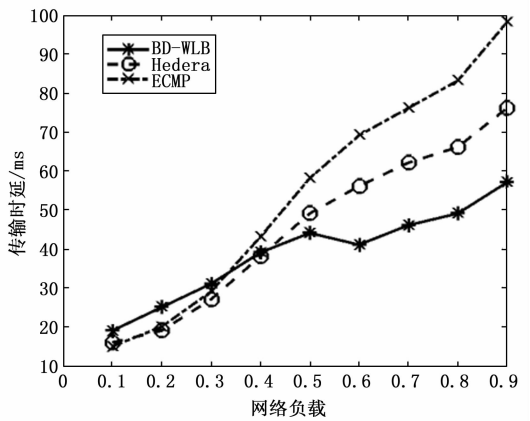

图 13 load _ x 模型网络时延

4 结束语

网络中不同特征的流量对于链路的需求不同，其中大象流对链路带宽条件敏感，老鼠流对链路时延条件敏感。因此本文设计的负载均衡算法的中心思想是根据大象流和老鼠流各自的敏感条件来选择最优的路径进行传输，从而减少网络拥塞的出现和排队时延过长等问题。通过控制器来进行全网状态信息的获取以及大象流检测，然后再根据网络状态信息计算大象流和老鼠流的最优转发路径，通过 P4 语言对数据平面转发流程进行优化，最终完成负载均衡。实验结果表明，相比于 ECMP 和 Hedera 算法，BD-WLB 算法有效提升了网络吞吐量、链路利用率、网络时延方面的性能，证明了算法的可行性和有效性。

但 BD-WLB 算法目前对于链路的评判标准还只是依靠网络带宽和链路时延两个参数。因此下一步的优化工作是为判断标准引入例如转发跳数等更多参数，进一步分析不同特征流量的需求，更真实的模拟网络状态，从而使算法更加适应真实网络。

参考文献：

[1] 谢添丞，吴凌淳，林羽丰，等. 面向大数据的网络流量监控与分析算法综述 [J]. 无线电通信技术，2022，48（5）：782

-793.
[2] 杨鹏飞. 基于 Kubernetes 的资源动态调度的研究与实现 [D]. 杭州：浙江大学，2017.
[3] 高恩池. 面向 IoT 超密集场景的接入及负载均衡技术研究 [D]. 成都：电子科技大学，2018.
[4] LIU Y，ZENG Z，LIU X，et al. A Novel Load Balancing and Low Response Delay Framework for Edge-Cloud Network Based on SDN [J]. IEEE Internet of Things Journal，2020，7 (7)：5922-5933.
[5] ALKHATIB A A，SAWALHA T，AlZU'BI S. Load Balancing Techniques in Software-Defined Cloud Computing：an overview [C] // 2020 Seventh International Conference on Software Defined Systems (SDS)，2020，240-244.
[6] MBAREK F，MOSOROV V. Load Balancing Based on Optimization Algorithms：An Overview [J]. Journal of Telecommunications and Information Technology，2019 (4)：3-12.
[7] HOPPS C. Analysis of an Equal-Cost Multi-Path Algorithm [S]. RFC 2992，IETF，2000.
[8] AL-FARES M，RADHAKRISHNAN S，RAGHAVAN B，et al. Hedera：dynamic flow scheduling for data center networks [C] // Proceedings of the 7th USENIX Symposium on Networked Systems Design and Implementation，NSDI 2010：28-30.
[9] ZAHER M，ALAWADI A H，MOLNAR S. Sieve：A flow scheduling framework in SDN based data center networks [J]. Computer Communications. 2021，171：99-111.
[10] LI Y，PAN D. OpenFlow based load balancing for Fat-Tree networks with multipath support [C] // Proceedings of the 12th IEEE International Conference on Communications (ICC513)，Budapest，Hungary. 2013：1-5.
[11] HONG C Y，CAESAR M，GODFREY P B. Software Defined Transport：Flexible and Deployable Flow Rate Control [C] //Proceedings of die ONS，2014：1-2.
[12] LIU Z，GAO D，LIU Y，et al. An Enhanced Scheduling Mechanism for Elephant Flows in SDN-Based Data Center [C] //Proceedings of the Vehicular Technology Conference (VTC-Fall)，IEEE，2016：1-5.
[13] 史久根，郑　伟，贾坤荥，等. 软件定义网络中基于负载均衡的多控制器部署算法 [J]. 电子与信息学报，2018，40 (2)：455-461.
[14] 曾志豪. 基于多控制器架构下的 SDN 网络关键技术研究 [D]. 成都：电子科技大学，2022.
[15] 赵文文，孟相如，康巧燕，等. 时延和可靠性感知的多控制器均衡部署策略 [J]. 空军工程大学学报 (自然科学版)，2021，22 (4)：85-91.
[16] BOSSHART P，GIBB G，KIM H S，et al. Forwarding metamorphosis：Fast programmable match-action processing in hardware for SDN [J]. ACM SIGCOMM Computer Communication Review，ACM，2013，43 (4)：99-110.
[17] BOSSHART P，DALY D，GIBB G，et al. P4：Programming protocol-independent packet processors [J]. ACM SIGCOMM Computer Communication Review，2014，44 (3)：87-95.
[18] NAGA K，HIRA M，KIM C，et al. HULA：Scalable Load Balancing Using Programmable Data Planes [C] //Proceedings of the Symposium on SDN Research. USA，2016，1-12.
[19] MIAO R，ZENG H，KIM C，et al. SilkRoad：Making Stateful Layer-4 Load Balancing Fast and Cheap Using Switching ASICs [C] //Proceedings of the Conference of the ACM Special Interest Group on Data Communication，USA，2017，15-28.
[20] OLTEANU V，AGACHE A，VOINESCU A，et al. Stateless datacenter load-balancing with beamer [C] //Proceedings of the Networked Systems Design and Implementation，USA，2018，18：125-139.
[21] HANDLEY M，RAICIU C，AGACHE A，et al. Re-architecting datacenter networks and stacks for low latency and high performance [C] //Proceedings of the Conference of the ACMS pecial Interest Group on Data Communication，SaintLouis，USA，2017：29-42.
[22] KANDULA S，SENGUPTA S，GREENBERG A，et al. The nature of data center traffic：measurements & analysis [C] // Proc of the 9th ACM SIGCOMM Conference on Internet Measurement，New York：ACM Press，2009：202-208.
[23] ALIZADEH M，GREENBERG A，MALTZ D A，et al. Data center TCP (DCTCP) [J]. ACM SIGCOMM Computer Communication Review，2010，40 (4)：63-74.
[24] FEAMSTER N，REXFORD J，ZEGURA E. The road to SDN：an intellectual history of programmable networks [J]. ACM SIGCOMM Computer Communication Review，2014，44 (2)：87-98.

(上接第 256 页)

[14] 李梦静，吉根林，赵　斌. 基于步行周期聚类的视频行人重识别关键帧提取算法 [J]. 南京航空航天大学学报，2021，53 (5)：780-788.
[15] 马境远，王川铭. 一种多尺度光流预测与融合的实时视频插帧方法 [J]. 小型微型计算机系统，2021，42 (12)：2567-2571.
[16] 梁恩泽，李宏刚，杜　双. 复杂机场全景视频拼缝优化方法研究 [J]. 电视技术，2021，45 (9)：119-123，128.
[17] 张哲蓄，孙立峰. 全景视频视野外关键信息感知技术 [J]. 中国科技论文，2021，16 (11)：1155-1161.
[18] 邢　丹，赵海武，滕国伟. 一种最小变形度全景视频映射方法 [J]. 电视技术，2021，45 (1)：17-24.
[19] 赵海武，陈　钰，吴成家，等. 全景视频最小变形双极方形映射研究 [J]. 计算机应用与软件，2021，38 (10)：139-143，152.
[20] 刘　煦，李　琛，宋　利，等. 基于 3D 旋转模型的全景视频稳像算法 [J]. 计算机应用与软件，2021，38 (6)：166-169，261.
[21] 曾　婷，黄东军. 智能视频监控系统异常行为检测算法研究综述 [J]. 计算机测量与控制，2021，29 (7)：1-6.