

## 一种基于动态配额的虚拟网带宽公平调度算法

刘中金<sup>①</sup> 卓子寒<sup>①</sup> 何跃鹰\*<sup>①</sup> 李勇<sup>②</sup> 苏厉<sup>②</sup> 金德鹏<sup>②</sup> 曾烈光<sup>②</sup>

<sup>①</sup>(国家计算机网络应急技术处理协调中心 北京 100029)

<sup>②</sup>(清华大学电子工程系 北京 100084)

**摘要:** 网络虚拟化被广泛用于网络实验平台和数据中心等场景中。作为虚拟化网络中的核心组网设备, 虚拟路由器可以在同一物理底层上构建多个虚拟路由器实例来承载多个虚拟网。其核心调度问题在于如何根据不同虚拟网对带宽的不同需求, 将网络数据包调度到不同的实例中。该文针对该问题对虚拟化场景下的队列调度问题进行建模, 提出了基于动态配额的队列调度算法, 与 miDRR 等算法相比, 该文算法在虚拟网带宽分配的有效性和公平性上有明显优势。

**关键词:** 网络虚拟化; 虚拟路由器; 软件定义网络; 队列调度算法; 公平性

中图分类号: TP393.1

文献标识码: A

文章编号: 1009-5896(2016)10-2654-06

DOI: 10.11999/JEIT151485

## Dynamical Weighted Scheduling Algorithm Supporting Fair Bandwidth Allocation of Virtual Networks

LIU Zhongjin<sup>①</sup> ZHUO Zihan<sup>①</sup> HE Yueying<sup>①</sup> LI Yong<sup>②</sup>  
SU Li<sup>②</sup> JIN Depeng<sup>②</sup> ZENG Lieguang<sup>②</sup>

<sup>①</sup>(National Computer Network Emergency Response Technical Team, Beijing 100029, China)

<sup>②</sup>(Department of Electronic Engineering, Tsinghua University, Beijing 100084, China)

**Abstract:** Network virtualization is widely deployed in network experiment platforms and data center networks. As a key networking equipment in virtualized environment, the virtual router can build many virtual router instances to run different virtual networks. The key problem for a virtual router lies in how to schedule the packets into different virtual instances according to the virtual networks' bandwidth requirement. In this article, a model is given to the scheduling problem and a dynamical weighted scheduling algorithm is proposed. The experimental results show that the proposed algorithm has superiority over miDRR algorithm in terms of the efficiency and the fairness.

**Key words:** Network virtualization; Virtual router; Software Defined Networking (SDN); Queue scheduling algorithm; Fairness

### 1 引言

软件定义网络(Software Defined Networking, SDN)<sup>[1]</sup>, VXLAN<sup>[2]</sup>等新协议不断涌现, 为了支持这些应用的部署, 研究人员引入了网络虚拟化的机制对网络进行隔离。在实验平台中, 管理员通过划分虚拟网实现不同协议、转发机制以及服务质量的分配和隔离<sup>[3]</sup>; 在数据中心中, 管理员将不同租户划分到不同的虚拟网, 以进行隔离和服务质量区分<sup>[4]</sup>。在这些场景中, 虚拟路由器作为核心的组网设备, 可

以在同一物理底层上构建多个虚拟路由器实例来承载多个虚拟网<sup>[5,6]</sup>, 因此, 管理员需要保证路由器能够按照虚拟网的服务质量要求对数据包进行队列调度。

队列调度算法已经得到了广泛而深入的研究, 算法从原理上可以分为两大类: 基于虚拟时间的算法, 如 WFQ(Weighted Fair Queuing)<sup>[7]</sup>, WF<sup>2</sup>Q(Worst-case Fair Queuing)<sup>[8]</sup>, STFQ(Start Time Fair Queuing)<sup>[9]</sup>, MS-PGPS<sup>[10]</sup>和 HFOB\_RSA<sup>[11]</sup>等算法和基于轮询的算法, 如差额轮询算法(Deficit Round Robin, DRR)<sup>[12]</sup>, SRR<sup>[13]</sup>, PDRR<sup>[14]</sup>和 miDRR(Multi-interface Deficit Round Robin)<sup>[15]</sup>等。

然而, 虚拟化网络环境中的队列调度问题与传

收稿日期: 2015-12-29; 改回日期: 2016-05-26; 网络出版: 2016-07-15

\*通信作者: 何跃鹰 hyy@cert.org.cn

基金项目: 国家高技术研究与发展计划(2012AA012801)

Foundation Item: The National High Technology Research and Development Program of China (2012AA012801)

统场景不同，它具有如下 3 个特点：

(1)虚拟网业务具有差异性。一个物理网络承载了多张并行的虚拟网，这些虚拟网通常属于不同的用户。用户会在虚拟网中部署自己的业务，使得不同虚拟网承载不同种类、不同数量的业务流。例如在实验平台和数据中心中，既有基于 IP 协议的业务，也会有基于 NDN<sup>[16]</sup>、OSA<sup>[17]</sup>、Avalanche<sup>[18]</sup>等新协议的业务。

(2)支持异构网络的数据平面虚拟化。在虚拟化的网络数据平面中，物理设备被虚拟化为多个虚拟转发实例。为了支持不同种类的业务，这些虚拟转发实例会被配置成不同的转发功能，处理不同格式的数据包。

(3)虚拟网业务流在数据平面中的处理具有选择性。受限于虚拟转发实例的处理类型，不同种类的业务流必须在不同种类的虚拟转发实例中处理。

当流与虚拟转发实例具有选择性时，虚拟完成时间将取决于虚拟转发实例上业务流的到达情况，不能仅通过当前时刻的队列状态决定数据包调度的先后顺序，因此基于虚拟完成时间的调度算法不适用于上述场景；在虚拟网和虚拟转发实例具有选择性的情况下，分布式的 DRR 的调度方案难以满足全局的业务流带宽分配需求；同时，以流为单位的 max-min 公平调度难以满足虚拟网的带宽分配要求。因此需要研究适用于虚拟网的调度算法，使得虚拟网之间的带宽得到公平分配。

## 2 虚拟化场景建模

本节根据虚拟化场景的特点对队列调度算法进行建模。如图 1 所示，不失一般性，考虑物理网络中的一个路由器  $R$ ，它被虚拟化为  $L$  个虚拟转发实例  $\{VR_j | 1 \leq j \leq L\}$ 。物理设备  $R$  上承载了  $N$  个业务流  $\{F_i | 1 \leq i \leq N\}$ ，这些业务流属于  $M$  个虚拟网  $\{VN_k | 1 \leq k \leq M\}$ 。业务流与虚拟网的归属关系用  $\Theta = [\theta_{ik}]$  表示，如果流  $F_i$  属于虚拟网  $VN_k$ ，那么  $\theta_{ik} = 1$ ，否则  $\theta_{ik} = 0$ 。虚拟网  $VN_k$  与虚拟转发实例  $VR_j$  之间的选择关系用流选择矩阵  $\Pi = [\pi_{kj}]$  来表示，如果虚拟网  $VN_k$  可以在虚拟路由器  $VR_j$  中处理，则  $\pi_{kj} = 1$ ，否则  $\pi_{kj} = 0$ 。

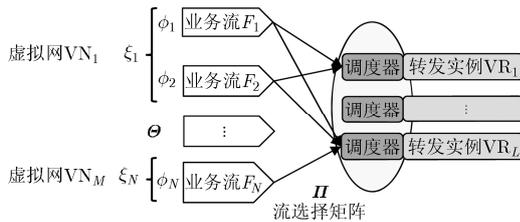


图 1 虚拟化场景中队列调度模型

## 3 基于动态配额的队列调度算法

在虚拟化网络中，虚拟网带宽 max-min 带宽公平是指：若要增加一个高带宽虚拟网的带宽，必须要降低一个低带宽虚拟网的带宽，即 max-min 公平分配保证了低带宽虚拟网所分配的带宽尽可能高。

本节根据上节所述的队列调度模型，提出基于动态配额的队列调度算法。下文将调度算法分为 3 部分进行描述，算法 1 是所提出算法的框架，算法 2 和算法 3 是调度算法的子算法，分别实现流选择和配额更新的功能。如图 1 所示，每个虚拟转发实例  $VR_j$  入口处部署一个调度器，在每个调度器上不断循环运行算法 1 中的基于动态配额的队列调度算法。算法所用符号及意义在表 1 中列出。

表 1 队列调度算法中所需符号意义

符号	具体涵义
$BL_i$	流 $F_i$ 的队列长度
$S_i$	流 $F_i$ 的队头数据包长度
$QI_i$	流 $F_i$ 的初始化配额
$Q_i$	流 $F_i$ 的配额
$DC_i$	流 $F_i$ 的赤字计数器
$B$	所有队列非空的流集合
$\xi_k$	虚拟网 $VN_k$ 的权重值
$B_k$	虚拟网 $VN_k$ 中队列非空的流集合
$F_j$	虚拟转发实例 $VR_j$ 服务的流集合
$C_j$	虚拟转发实例 $VR_j$ 当前服务的流
$SF_{ij}$	虚拟路由器 $VR_j$ 对流 $F_i$ 的服务标志

以虚拟转发实例  $VR_j$  为例，每次调度分为 3 个阶段，(1)流选择阶段，在每次调度开始时，调度器首先判断虚拟转发实例  $VR_j$  是否有业务流排队，如果没有流排队，那么结束本次调度；如果有业务流处于排队状态，那么队列指针  $i$  指向下一个需要服务的流队列  $C_j$ ；(2)队列服务阶段，如果差额计数器  $DC_i$  大于当前服务队列队头的数据包长度  $S_i$ ，则从计数器中减掉队头数据包长度，并将队头数据包发送到  $VR_j$ ，进入调度决策阶段，否则，直接进入调度决策阶段；(3)调度决策阶段，如果当前服务队列为空  $BL_i = 0$ ，就将该队列的差额计数器置零，并将该流从  $B$  中删除。如果当前服务队列为空  $BL_i = 0$  或者差额计数器  $DC_i$  小于队头的数据包长度  $S_i$ ，那么按照算法 2 将队列指针  $i$  指向下一个需要服务的队列，并按照算法 3 更新业务流的配额，加到流的差额计数器  $DC_i$  上，调度决策阶段结束。

因此，算法 1(表 2)的每次循环至多发送一个数据包，每发送一次进行判断，如果仍然有差额计数器  $DC_i$  大于队头的数据包长度  $S_i$ ，指针  $i$  保持不变，

表2 基于动态配额的队列调度算法

算法1 基于动态配额的队列调度算法(虚拟实例  $VR_j$  的调度算法)

- ```

(1)      IF  $F_j \cap B = \emptyset$  THEN
(2)          Return
(3)      END IF
(4)       $i = C_j$ 
(5)      IF  $S_i \leq DC_i$  THEN
(6)          发送长度为  $S_i$  的数据包
(7)           $DC_i = DC_i - S_i$ 
(8)      END IF
(9)      IF  $BL_i = 0$  THEN
(10)          $DC_i = 0$ 
(11)         将  $F_i$  从集合  $B$  中删除
(12)     END IF
(13)     IF  $BL_i = 0$  or  $S_i > DC_i$  THEN
(14)          $i = C_j = VR_j$  下一个排队的流(算法2)
(15)          $DC_i = DC_i + Q_i$  (算法3)
(16)     END IF

```

并进入下一次循环发送数据包,直到  $DC_i$  小于队头的数据包长度  $S_i$ 。经过多次循环,调度器指针  $i$  可以将  $VR_j$  所有可服务的队列遍历一遍,称为一轮调度。

算法2(表3)引入单比特服务标志变量  $SF_{ij}$  实现多个调度器的服务信息共享。调度器中维护了一个矩阵,存储每一个虚拟转发实例  $VR_j$  对每一条业务流  $F_i$  的服务标志  $SF_{ij}$ 。在调度器初始化时刻,所有的服务标志  $SF_{ij}$  都为0,在调度过程中,如果某条流  $F_i$  被虚拟转发实例  $VR_j$  服务过,那么服务标志  $SF_{ij}$  会被置为1,而其他所有的虚拟转发实例对流  $F_i$  的服务标志会被置为0,即  $SF_{im} = 1, \forall m \neq j$ 。基于这一机制,算法2包括了两个阶段:(1)流选择过程,指针  $i$  指向  $VR_j$  中下一个排队的流,判断服务标志  $SF_{ij}$ ,如果发现该标志值为1,说明本轮调度中流  $F_i$

表3 流选择算法

算法2 流选择算法

- ```

(1)       $i = C_j = VR_j$  下一个排队的流
(2)      WHILE  $SF_{ij} \neq 0$  DO
(3)           $SF_{ij} = 0$ 
(4)           $i = C_j = VR_j$  下一个排队的流
(5)      END WHILE
(6)       $SF_{im} = 1, \forall m \neq j$ 
(7)      Return( $i$ )

```

已被其他虚拟转发实例服务过,那么  $VR_j$  就跳过该流,并将  $SF_{ij}$  重置为0,用于下一轮调度;接着更新指针  $i$  指向下一个排队的流,判断服务标志,如此不断循环,直到找到  $SF_{ij} = 0$  的流,即没有被其他虚拟转发实例服务过的流,跳出循环。(2)流服务标志修改,找到  $SF_{ij} = 0$  的流后,  $VR_j$  对该流  $F_i$  进行服务,并将其他所有的虚拟转发实例对应流  $F_i$  的服务标志置为1,即通知其他虚拟转发实例在本轮调度中不要服务  $F_i$ 。这样,每轮调度中  $VR_j$  只会服务未被其他虚拟转发实例服务过的流。从全局来看,在所有调度器的每轮调度过程中,每条流最多会被一个虚拟转发实例处理一次,保证了流调度的公平性。

在已有算法中调度器以流为单位进行队列调度,流的配额是一成不变的。算法3(表4)考虑了流与虚拟网的归属关系,根据虚拟网中活跃业务流的数目在每一轮调度中更新所有流的配额,使得虚拟网内的各条业务流则按照比例对配额  $Q_i$  进行调整,同时每轮调度中每个虚拟网的总配额  $\xi_k$  保持不变。

表4 动态配额更新算法

算法3 动态配额更新算法

- ```

(1)       $VN_k =$  流  $F_i$  所属于的虚拟网
(2)       $\xi = \frac{\xi_k}{\sum_{i \in B_k} Q_i}$ 
(3)      IF  $BL_i \neq 0$  THEN
(4)           $Q_i = Q_i \cdot \xi$ 
(5)      ELSE THEN
(6)           $Q_i = 0$ 
(7)      END IF
(8)      Return( $Q_i$ )

```

可以证明基于算法1的调度器满足速率集簇特性,因此该调度器能够实现虚拟网之间带宽的 max-min 公平调度。

## 4 性能评估

### 4.1 实验设计

为了评估本文所提队列调度算法的有效性和公平性,我们基于 NetFPGA 硬件板卡进行了验证<sup>[10]</sup>。我们在单个 NetFPGA 板卡上创建了2个虚拟转发实例  $VR_1$  和  $VR_2$ ,二者的处理容量都为1000 Mbps。通过流量产生器发送3个TCP业务流  $F_1, F_2, F_3$  到板卡上,其中  $F_1$  和  $F_2$  属于虚拟网  $VN_1$ ,可以在  $VR_1$  和  $VR_2$  中处理;  $F_3$  属于虚拟网  $VN_2$ ,仅能在  $VR_2$  中

处理。虚拟网的带宽权重为  $\xi_1 : \xi_2 = 3 : 1$ ，业务流的带宽权重为  $\varphi_1 : \varphi_2 : \varphi_3 = 1 : 3 : 1$ 。作为比较，在 VR<sub>1</sub> 和 VR<sub>2</sub> 的调度器中分别部署了 miDRR 和算法 1，为了能够反映调度算法的动态特征，在 VR<sub>1</sub> 和 VR<sub>2</sub> 中统计了 10 s 的流量，其中 F<sub>1</sub> 和 F<sub>3</sub> 持续了 10 s，F<sub>2</sub> 持续了 4 s，下面对实验结果进行分析。

### 4.2 算法有效性测试

在所设计实验条件场景下，DRR, SRR 和 PDRR 调度算法为各个数据流分配的带宽是一致的，其主要区别在于算法复杂度和调度延时上。

图 2(a<sub>1</sub>)和图 2(a<sub>2</sub>)分别示出了在上述 3 种算法下 VR<sub>1</sub> 和 VR<sub>2</sub> 内业务流的带宽分配情况。由于 VR<sub>1</sub> 和 VR<sub>2</sub> 的调度过程相互独立，前 4 s，VR<sub>1</sub> 中 F<sub>1</sub> 和 F<sub>2</sub> 两个流的占比为 1:3，在 VR<sub>2</sub> 中 F<sub>1</sub>，F<sub>2</sub> 和 F<sub>3</sub> 的比例为 1:3:1；在 4~6 s 间，到 6 s 后，F<sub>2</sub> 队列不再有数据包排队，在 VR<sub>1</sub> 中只处理 F<sub>1</sub>，而在 VR<sub>2</sub> 中，F<sub>1</sub> 和 F<sub>3</sub> 按照  $\varphi_1 : \varphi_3 = 1 : 1$  的比例进行分配。

图 2(b<sub>1</sub>)和图 2(b<sub>2</sub>)分别示出了在 miDRR 算法下 VR<sub>1</sub> 和 VR<sub>2</sub> 内业务流的带宽分配情况。从图 2(b<sub>1</sub>)可以看出，在前 4 s，F<sub>1</sub> 和 F<sub>2</sub> 会被交替调度到 VR<sub>1</sub> 和 VR<sub>2</sub> 中，F<sub>3</sub> 则严格地在 VR<sub>2</sub> 中处理；在 4~6 s 间，由于 F<sub>2</sub> 停止到达，因此 VR<sub>1</sub> 会处理队列中剩余的数据包，其处理速率也逐渐下降；在 6 s 后，F<sub>2</sub> 队列不再有数据包排队，F<sub>1</sub> 和 F<sub>3</sub> 分别在 VR<sub>1</sub> 和 VR<sub>2</sub> 中满速率处理，符合  $\varphi_1 : \varphi_3 = 1 : 1$  的比例。

图 2(c<sub>1</sub>)和图 2(c<sub>2</sub>)分别示出了在本文算法下

VR<sub>1</sub> 和 VR<sub>2</sub> 内业务流的带宽分配情况。可以看到，前 4 s，F<sub>1</sub> 和 F<sub>2</sub> 仍然在两个虚拟实例中处理；在 6 s 后，F<sub>2</sub> 队列空置，可以看出，VR<sub>1</sub> 和 VR<sub>2</sub> 对 F<sub>1</sub> 和 F<sub>2</sub> 的配额进行调整，F<sub>3</sub> 的配额不变。因此，在 VR<sub>1</sub> 和 VR<sub>2</sub> 中 F<sub>1</sub> 的带宽分别增长到 1000 Mbps 和 500 Mbps，F<sub>3</sub> 的带宽保持不变。

图 3(a<sub>1</sub>)和图 3(a<sub>2</sub>) 显示了 DRR, SRR 和 PDRR 调度算法下，3 个流和虚拟网的总体带宽的分配情况。从图 3(a<sub>1</sub>)中可以看出：前 4s，3 条流的带宽分配分别为 450 Mbps, 1350 Mbps 和 200 Mbps，不符合预期的 1:3:1 的流权重分配。从图 3(a<sub>2</sub>)中可以看出，VN<sub>1</sub> 与 VN<sub>2</sub> 的带宽比例为 9:1，未按照预定的虚拟网权重进行调度。从第 6 s 开始，F<sub>2</sub> 不再有数据包排队，其配额变为 0，VN<sub>1</sub> 与 VN<sub>2</sub> 的带宽比例变为 3:1，也未能按照虚拟网权重进行调度。

图 3(b<sub>1</sub>)和图 3(b<sub>2</sub>)显示了 miDRR 调度算法下，3 个流和虚拟网的总体带宽的分配情况。从图 3(b<sub>1</sub>)中可以看出：前 4 s，3 条流的带宽分配分别为 400 Mbps, 1200 Mbps 和 400 Mbps，符合预期的 1:3:1 的流权重分配。从图 3(b<sub>2</sub>)中可以看出，VN<sub>1</sub> 与 VN<sub>2</sub> 的带宽比例为 4:1，未按照预定的虚拟网权重进行调度。从第 6 s 开始，F<sub>2</sub> 不再有数据包排队，其配额变为 0，因此，F<sub>1</sub> 和 F<sub>3</sub> 按照  $\varphi_1 : \varphi_3 = 1 : 1$  比例分别在 VR<sub>1</sub> 和 VR<sub>2</sub> 中处理，VN<sub>1</sub> 与 VN<sub>2</sub> 的带宽比例也是 1:1，也未能按照虚拟网权重进行调度。

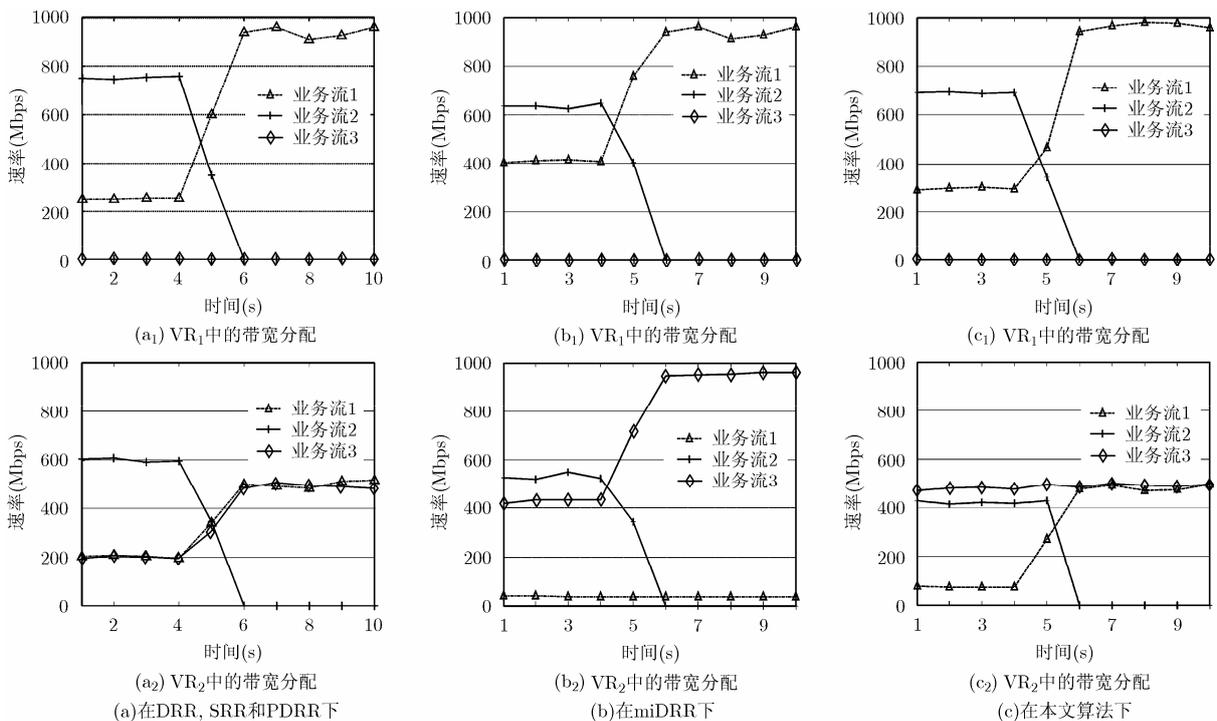


图 2 DRR, miDRR 和本文算法在虚拟转发实例中的带宽分配情况

图 3(c<sub>1</sub>)和图 3(c<sub>2</sub>)显示了在本文所提调度算法下, 3 个流和虚拟网的总体带宽的分配情况。在前 4 s, 3 条流的速率分别为 370 Mbps, 1130 Mbps 和 500 Mbps, VN<sub>1</sub> 与 VN<sub>2</sub> 的带宽比例为 3:1, 与预期权重相同, 同时, 在 VN<sub>1</sub> 内部, F<sub>1</sub> 和 F<sub>2</sub> 保持了 1:3 的比例。4 s 后, 当 F<sub>2</sub> 的带宽下降时, 调度器调整 F<sub>1</sub> 的配额, 使得 F<sub>1</sub> 的带宽分配迅速增加并保持在 1500 Mbps, 从而保证了虚拟网 VN<sub>1</sub> 与 VN<sub>2</sub> 的带宽维持在 3:1。说明本文算法能够以虚拟网为单位调度业务流。

### 4.3 算法公平性测试

为了测试队列调度算法的公平性, 我们将带宽比例 VN<sub>1</sub> 与 VN<sub>2</sub> 的带宽比例需求从 1:10 逐渐变化为 10:1。我们分别测试了不同比例下的带宽分配情况, 如图 4(a)和图 4(b)所示, 二者分别显示了不同

比例区间范围内的带宽分配情况, 其中实线为按照虚拟网 max-min 公平分配的理论值, 十字虚线为按照比例分配的理论值, 三角虚线为带宽实际测量值。从图 4(a)可以看出在所提调度算法下, 带宽分配情况与 max-min 公平分配的理论值基本相符。从图 4(b)可以看出, 当 VN<sub>1</sub> 与 VN<sub>2</sub> 的带宽分配比例低于 1:1 时, 带宽能够按照 max-min 公平分配, 为每个虚拟网都会分配 1000 Mbps 的带宽, 使得带宽维持在 1:1 的比例。当 VN<sub>1</sub> 与 VN<sub>2</sub> 带宽比例超过 1:1, 带宽按照预先配置的比例进行分配, 说明所提算法能够提供 max-min 公平的带宽分配。

## 5 结束语

针对虚拟化环境中虚拟网之间带宽分配的问题, 本文对虚拟路由器的队列调度问题进行了建模,

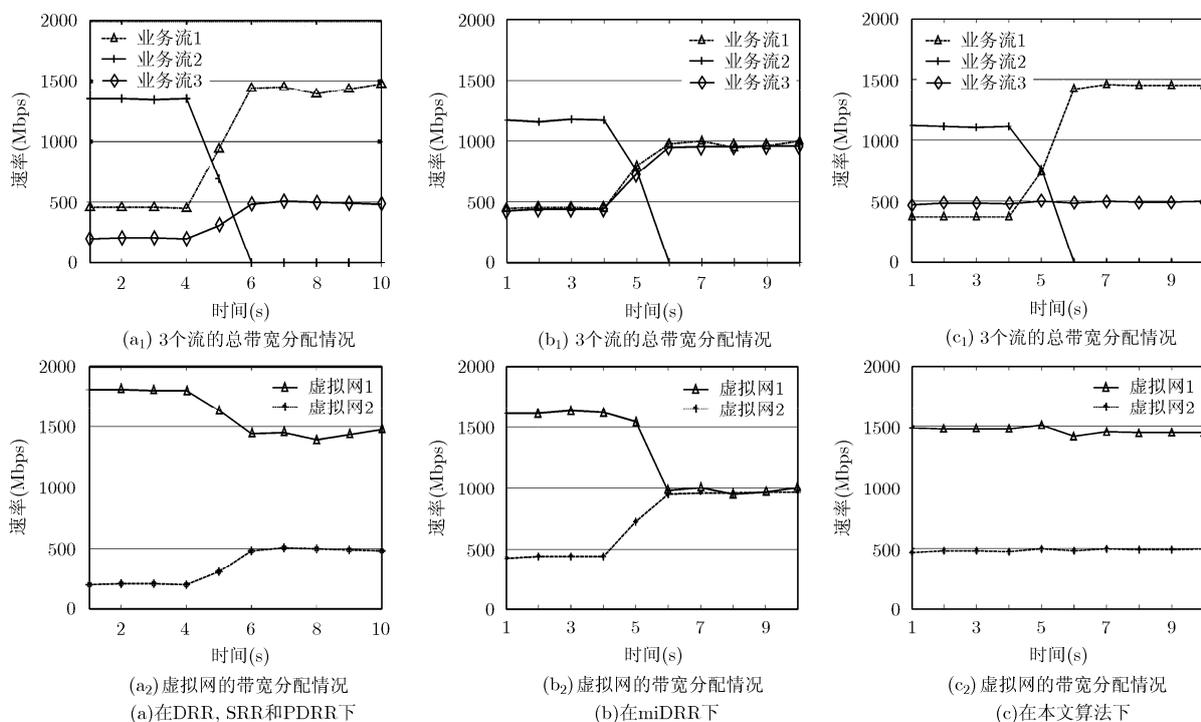


图 3 DRR, miDRR 和本文算法在业务流及虚拟网总体带宽分配情况

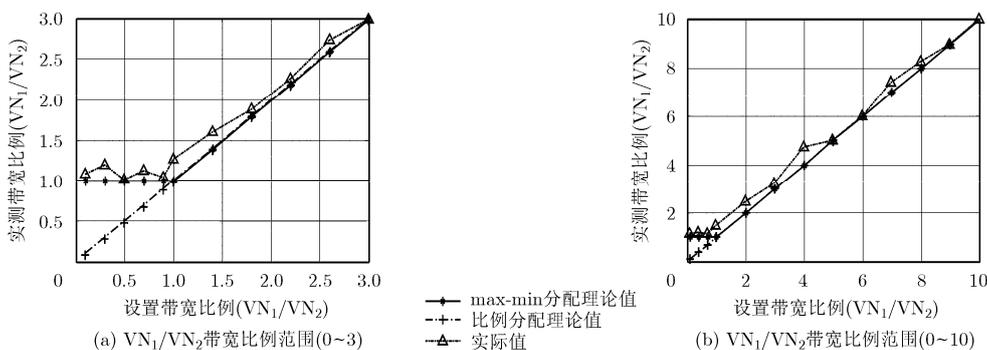


图 4 不同虚拟网带宽比例设置下的带宽实际分配情况

并提出了基于动态配额的队列调度算法。本文算法能够以虚拟网为单位进行带宽资源的分配,与 DRR, SRR 和 miDRR 等算法相比,本文算法在虚拟网带宽分配的有效性和公平性上有显著优势。

### 参 考 文 献

- [1] KREUTZ D, RAMOS F M V, ESTEVES V P, *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] MAHALINGAM M, DUTT D, DUDA K, *et al.* Virtual extensible local area network (VXLAN): a framework for overlaying virtualized layer 2 networks over layer 3 networks [R]. 2014.
  - [3] BERMAN M, CHASE J S, LANDWEBER L, *et al.* GENI: a federated testbed for innovative network experiments[J]. *Computer Networks*, 2014, 61: 5-23. doi: 10.1016/j.bjp. 2013.12.037.
  - [4] KOPONEN T, AMIDON K, BALLAND P, *et al.* Network virtualization in multi-tenant data centers[C]. Proceedings of the 11th USENIX Symposium on Networked Systems Design and Implementation, Seattle, USA, 2014: 203-216.
  - [5] 刘中金, 李勇, 杨懋, 等. 基于可编程硬件的虚拟路由器数据平面设计与实现[J]. *电子学报*, 2013, 41(7): 1268-1272. doi: 10.3969/j.issn.0372-2112.2013.07.004.  
LIU Zhongjin, LI Yong, YANG Mao, *et al.* Design on data plane of programmable hardware-based virtual router[J]. *Acta Electronica Sinica*, 2013, 41(7): 1268-1272. doi: 10.3969/j.issn.0372-2112.2013.07.004.
  - [6] 刘中金, 李勇, 苏厉, 等. 弹性协议可定制的网络数据平面结构及其映射算法[J]. *电子与信息学报*, 2014, 36(7): 1713-1719. doi: 10.3724/SP.J.1146.2013.01151.  
LIU Zhongjin, LI Yong, SU Li, *et al.* Design on the elastic protocol customizable data plane and its mapping algorithm[J]. *Journal of Electronics & Information Technology*, 2014, 36(7): 1713-1719. doi: 10.3724/SP.J.1146.2013.01151.
  - [7] PAREKH A K and GALLAGER R G. A generalized processor sharing approach to flow control in integrated services networks: the single-node case[J]. *IEEE/ACM Transactions on Networking*, 1993, 1(3): 344-357. doi: 10.1109/INFCOM.1992.263509.
  - [8] BENNETT J C R and ZHANG H. WF2Q: worst-case fair weighted fair queueing[C]. Proceedings of IEEE INFOCOM'96, 1996, Vol. 1: 120-128. doi: 10.1109/INFCOM.1996.497885.
  - [9] GOVAL P, VIN H M, and CHENG H. Start-time fair queueing: A scheduling algorithm for integrated services packet switching networks[J]. *IEEE/ACM Transactions on Networking*, 1997, 5(5): 690-704.
  - [10] BLANQUER J M and ÖZDEN B. Fair queueing for aggregated multiple links[J]. *ACM SIGCOMM Computer Communication Review*, 2001, 31(4): 189-197. doi: 10.1145/383059.383074.
  - [11] 高先明, 张晓哲, 王宝生, 等. 面向虚拟路由器的基于历史转发开销的资源调度算法[J]. *电子与信息学报*, 2015, 37(3): 686-692. doi: 10.11999/JEIT140491.
  - [12] SHREEDHAR M and VARGHESE G. Efficient fair queuing using deficit round-robin[J]. *IEEE/ACM Transactions on Networking*, 1996, 4(3): 375-385. doi: 10.1109/90.502236.
  - [13] GUO C. SRR: an O(1) time complexity packet scheduler for flows in multi-service packet networks[J]. *ACM SIGCOMM Computer Communication Review*, 2001, 31(4): 211-222. doi: 10.1109/TNET.2004.838601.
  - [14] TSAO S C and LIN Y D. Pre-order deficit round robin: a new scheduling algorithm for packet-switched networks[J]. *Computer Networks*, 2001, 35(2): 287-305. doi: 10.1016/S1389-1286(00)00172-9.
  - [15] YAP K K, SANDEEP Y Y, and KATTI K S. Scheduling packets over multiple interfaces while respecting user preferences[C]. Proceedings of the Ninth ACM Conference on Emerging Networking Experiments and Technologies. Santa Barbara, 2013: 109-120. doi: 10.1145/2535372.2535387.
  - [16] VAN J, PATRICK C, ZHANG L, *et al.* Named data networking[OL]. <http://www.named-data.net/>, 2015, 12.
  - [17] CHEN K, SINGLA A, SINGH A, *et al.* OSA: an optical switching architecture for data center networks with unprecedented flexibility[J]. *IEEE/ACM Transactions on Networking*, 2014, 22(2): 498-511.
  - [18] IYER A, KUMAR P, and MANN V. Avalanche: data center multicast using software defined networking[C]. Proceedings of IEEE Sixth International Conference on Communication Systems and Networks (COMSNETS), Bangalore, India, 2014: 1-8. doi: 10.1109/COMSNETS.2014.6734903.
  - [19] LOCKWOOD J W, MCKEOWN N, WATSON G, *et al.* NetFPGA—an open platform for gigabit-rate network switching and routing[C]. Proceedings of the IEEE International Conference on Microelectronic Systems Education, San Diego, USA, 2007: 160-161. doi: 10.1109/MSE.2007.69.
- 刘中金: 男, 1988年生, 博士, 研究方向为计算机网络安全、软件定义网络、数据中心网络、可编程虚拟化路由器等。  
卓子寒: 男, 1987年生, 博士, 研究方向为计算机网络安全、网络测量、图像处理等。  
何跃鹰: 男, 1975年生, 高级工程师, 研究方向为大数据分析、网络安全。  
李勇: 男, 1985年生, 讲师, 研究方向为软件定义网络、下一代IP网络体系结构、移动容迟网络、网络虚拟化等。  
苏厉: 男, 1976年生, 讲师, 研究方向为片上网络、软件定义网络、短距离无线通信等。  
金德鹏: 男, 1972年生, 教授, 研究方向为软件定义网络、片上网络、短距离无线通讯等。  
曾烈光: 男, 1947年生, 教授, 研究方向为通信网、ASIC设计、片上网络、下一代网络体系架构等。