第21卷第3期　湖南工业大学学报　Vol.21 No.3
2007年5月　Journal of Hunan University of Technology　May 2007

FRACE流量控制机制的研究

曹玉军，高守平

（湘南学院 计算机科学系，湖南 郴州 423000）

摘　要：在对PAUSE功能以及其他研究分析和总结的基础上，提出了名为FRACE的可用于全双工以太网的流量控制机制，利用这种机制可以提高以太网的网络性能、保障用户服务质量。

关键词：以太网；流量控制；令牌桶

中图分类号：TP393.11　**文献标识码：**A　**文章编号：**1673-9833(2007)03-0042-06

Research on controlling Mechanism of FRACE Flow

Cao Yujun，Gao Shouping

（Department of Computer Science, XiangNan University, Chenzhou Hunan 423000, China）

Abstract：On the basis of the analysis and summarization of the function of PAUSE and other research, FRACE -a flow controlling mechanisms used in all duplex Ethernet is put forward, which can enhance the performance of Ethernet and ensure the service quality for the users.

Key words：Ethernet；flow control；token bucket

1　以太网流量控制需求

在以太网中，可能导致帧传输出错的原因有两个：数据错和丢帧。以太网中的数据出错概率非常小，10 Mb/s的铜介质中规定的最坏情况下的比特错误率为10~8[1]，在百兆和千兆以太网中，差错率更低。这种出错概率对于数据链路层已经低到可以忽略不计的程度，完全可以满足高层协议可靠数据传输的需要。因此，我们需要考虑的主要问题是丢帧导致的帧传输错。

近年来，随着运行于以太网上的用户和应用越来越多，数据量越来越大，帧到达的速度可能比交换机接收、处理和转发的速度要快，因此交换机在缓存溢出时不得不丢弃到来的帧，直到拥塞消除。这种丢帧对大多数可靠的传输层协议（TCP、SPX等）是非常有害的。因为这些协议都使用“确认重传”（PAR）算法[2]，用以实现可靠的端到端通信。这种机制可以很好地保证可靠的数据通信。然而，交换机的丢帧将导致协议的确认定时器超时，引起发送方重传丢失帧。而由于拥塞的消除通常需要一段时间，在此过程中会出现连续的丢帧，这种连续丢帧的状况将导致这些可靠的传输层协议有很大的性能下降[3]。

因此，在这样的情况下，仅仅依靠端到端流量控制机制在以太网中提供可靠数据传输是远远不够的。因此链路层丢帧的问题必须在链路层解决，这就是以太网流量控制提出的原因。

2　现有以太网流量控制机制及其不足

为了在全双工以太网中实现流量控制，IETF提出了IEEE 802.3x标准[4]。

2.1　MAC控制通用体系结构框架

IEEE 802.3x工作组定义了一个以太网MAC控制通用体系结构框架，MAC控制层是数据流链路层的一个子层，介于传统以太网MAC层和MAC层客户之间。MAC层客户可以是网络层协议（如IP）或在数据链路层内部实现转发功能的交换机。MAC控制层在OSI的

收稿日期：2007-04-05
作者简介：曹玉军（1967-），男，湖南永兴人，湘南学院讲师，主要研究方向为计算机网络与信息安全；
高守平（1965-），男，湖南常德人，湘南学院教授，博士，主要研究方向为网格计算.

7层参考模型中的位置如图1所示。

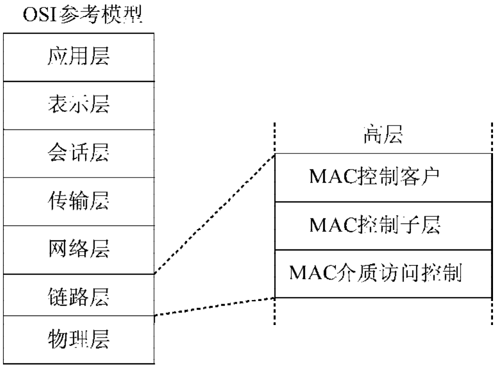

图1　MAC控制层图示图

Fig. 1　MAC control layer

如果MAC层客户不使用或不关心MAC控制层提供的功能，则是不可见的。而如果需要使用MAC控制层的功能，就可以利用该子层控制底层以太网MAC的操作，例如后面我们将要提到的用PAUSE功能实现以太网交换机的流控功能。MAC控制层的功能是通过MAC控制帧来实现的，它由数据链路层本身产生和接收，携带的是MAC层的控制信息。MAC控制帧符合标准的MAC帧格式，并用唯一的类型域标识符（0x8808）标识出。在控制帧的数据域内，前两个字节表示MAC控制的操作代码，即请求的控制功能，操作代码后面的域是该操作所需的参数。MAC控制帧格式如图2所示。MAC帧与数据帧之间的关系及控制帧使用过程如图3所示。

2.2　PAUSE流量控制

在MAC控制框架的基础上，IEEE 802.3x定义了用于在全双工以太网上实现流量控制的PAUSE功能。PAUSE功能利用MAC控制体系结构及帧格式实现，可用于在具有PAUSE功能的交换机或主机间实现流量控制。PAUSE帧包含了MAC控制帧的所有域，其目的地址是一个为PAUSE保留的唯一组播地址01-80-C2-00-00-01，源地址为发送者的MAC地址，类型域为0x8808。控制操作码是0x0001，并带有一个暂停时间参数，该参数是一个2字节的无符号整数，代表发送方请求接收方停止发送数据帧的时间长度，时间单位为以当前数据传输速率传送512比特的时间。PAUSE帧的结构如图4所示。

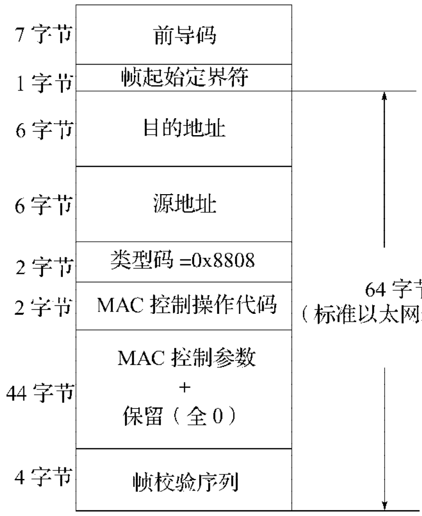

图2　MAC控制帧格式

Fig. 2　MAC control format

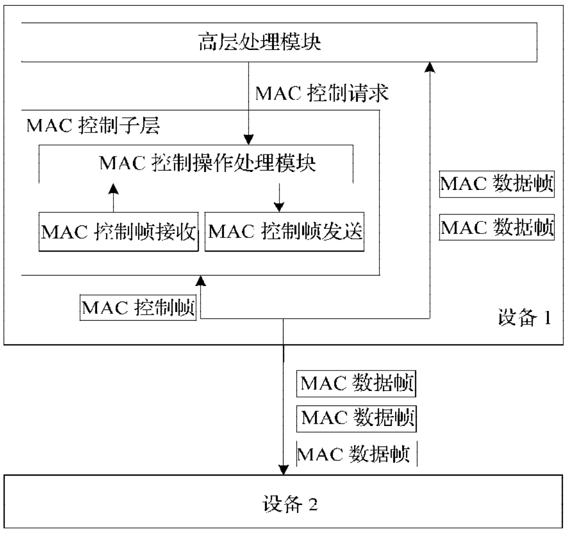

图3　MAC控制帧与数据帧过程

Fig. 3　MAC control frame and data process

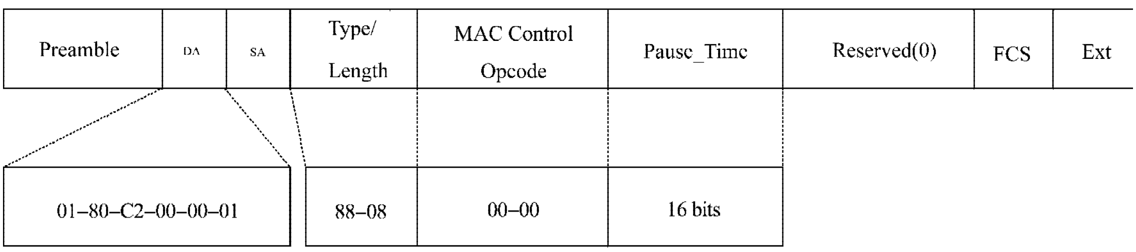

图4　PAUSE帧结构

Fig. 4　The structure of PAUSE frame

PAUSE功能的目的是为了在设备短时过载而导致缓冲区溢出时避免不必要的丢帧。PAUSE功能实现了一种简单的“停止/启动”形式的流量控制，其实现方法包括两个部分：拥塞检测和流量控制。PAUSE功能的操作过程如图5所示。

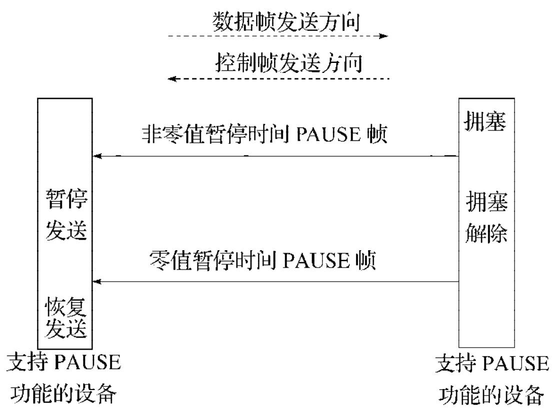

图5 PAUSE操作过程

Fig. 5 The operating process of PAUSE

PAUSE功能的定义决定了PAUSE功能不能解决以下问题：

1）稳定状态的网络拥塞：PAUSE功能的设计目标是在缓存区溢出时通过减少到达的数据帧，缓和短时过载的情况，并不考虑持续状态的网络拥塞，这种情况是网络设计或配置的问题，而不是流量控制所能解决的问题。

2）提供端到端的流量控制：PAUSE功能只实现逐跳的流量控制，不具备端到端的流量控制能力，这也不是PAUSE功能的设计目标。

3）提供比简单“停止/启动”更复杂的机制：不能实现基于流的策略和基于速率的流量控制等更复杂的操作。

3 FRACE流量控制机制

通过分析，我们知道，在目前复杂的以太网应用环境中，流量控制需要解决以下4个主要问题：提供基于流的流量控制能力；区分不同业务的数据帧；实现相对精确的流量控制；可以在设备中简单实现。

以上4个问题的解决方法如下：

1）基于流的流控：我们定义由源MAC地址和目的MAC地址确定的流，利用分类器，根据到达的数据帧帧头的地址参数进行分类，通过相应的速率控制器进行流量控制。

2）业务区分：在符合802.1Q协议的数据帧头中，包含了一个可用于标识业务类型的字段User_Priority[5]。我们可以在流量控制机制中引入该参数，用于对数据帧所属的业务类型进行区分。

3）精确流控：实践证明，令牌桶算法是实现精确流控的有效方法，因此可以采用令牌桶实现相对精确的流量控制。

4）实现简单：只是在MAC框架内定义了一个与PAUSE功能处于相同层次的FRACE（Flow RAte Control in Ethernet）流量控制功能。同时，基于MAC地址的流识别，基于User_priority字段的业务区分都是以太网交换设备已有的能力，不会增加交换机的额外负担。基于令牌桶的流控已有成熟的算法，可以在硬件中很方便地实现。

3.1 FRACE实现结构

综合上述4个方面的解决方法，我们设计出了FRACE流控的实现结构，如图6所示。

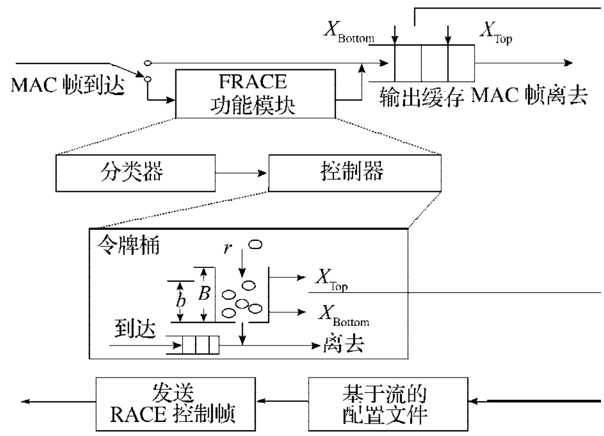

图6 FRACE实现结构图

Fig. 6 The implementation structure of FRACE

FRACE控制功能由集成到每个输出端口的FRACE功能单元完成。该功能单元可以由FRACE MAC控制帧使能或禁止。当FRACE功能单元被禁止时，到达的数据帧直接被放入输出缓存以备发送；当FRACE功能单元使能时，数据帧将通过该功能单元进行流量控制。

FRACE功能单元由以下两部分组成：

1）分类器：分类器的功能是根据数据帧头的3个字段将数据帧分发到相应的控制器中，这3个字段是源MAC地址、目的MAC地址和优先级。

2）控制器：控制器由一组令牌桶组成。令牌的生成速率r即为用户流量的控制速率。令牌桶容量B为用户最大突发长度。当数据帧到达控制器时，只有当令牌桶中当前令牌数量b大于数据帧长度时才能通过，然后减去相应长度的令牌数。

FRACE功能单元需要用到的具体参数包括：源MAC地址、目的MAC地址、优先级和控制速率。这些参数由配置文件定义，存储于设备中。参数可以通过设备控制界面或网管接口修改设置，如SNMP[6]。

在输出缓存区和每个令牌桶都设置有两个门限值：X_{Top}和X_{Bottom}，用于决定何时进行流量控制或解除流量控制，这两个门限值的设置与PAUSE功能定义的

X_{OFF} 和 X_{ON} 门限值类似。控制或解除控制操作分为两种情况：

1）当输出缓存中的数据帧总量超过 X_{Top} 时，说明输出端口发生拥塞，需要对设置的所有数据流进行流量控制。设备从配置文件中读取参数值，构造带有非零值控制速率参数的FRACE控制帧向上游节点发送。当输出缓存中的数据帧总量回落到 X_{Bottom} 以下时，说明输出端口的拥塞已解除，需要解除对上游节点进行的流量控制。设备构造带有最大控制速率参数值（代表解除所有流的流量控制）的FRACE控制帧向上游节点发送。

2）当某个令牌桶的令牌数量小于 X_{Bottom} 时，说明相应数据流的速率过大，需要进行流量控制。设备从配置文件中读取相应流的参数值，构造带有非零值控制速率参数的FRACE控制帧向上游节点发送。当令牌桶中的令牌数量上升到 X_{Top} 以上时，说明该数据流的速率已得到控制，可以解除上游节点对该数据流的流量控制。设备构造带有等于解除流控值的控制速率参数，以及相应流的MAC地址的FRACE控制帧向上游节点发送。

3.2　FRACE控制帧

FRACE控制帧是触发FRACE流量控制功能的特殊帧，它符合802.3x定义的MAC控制帧结构，如图7所示。

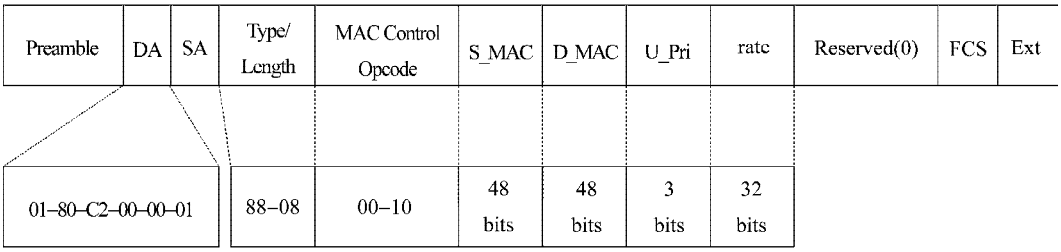

图7　FRACE控制帧结构

Fig. 7　The structure of FRACE control frame

FRACE控制帧包含了MAC控制帧的所有域。其目的地址是一个为与PAUSE帧相同的组播地址：01-80-C2-00-00-01，源地址为发送者的MAC地址。类型域为0x8808，控制操作码是0x0010。FRACE控制帧带有4个控制参数：

1）S_MAC：6字节的MAC地址，指定了需要进行流量控制的数据流的源MAC地址。该参数为FF-FF-FF-FF-FF-FF时，代表匹配所有的MAC地址。

2）D_MAC：6字节的MAC地址，指定了需要进行流量控制的数据流的目的MAC地址。该参数为FF-FF-FF-FF-FF-FF时，代表匹配所有的MAC地址。

3）U_Pri：3比特的用户优先级字段，指定了需要进行流量控制的数据帧的业务类型。

4）rate：4字节的无符号整数，指定流量控制的速率参数，单位为kbps，取值范围为0~10 Gbps。该参数为0xFFFFFFFE代表取消指定流的流量控制；该参数为0xFFFFFFFF代表取消所有流的流量控制。

3.3　FRACE流量控制过程

为说明FRACE流控的操作过程，我们以FRACE在如图8所示的网络环境中的应用为例，详细说明整个流控的处理过程，FRACE的流量控制过程见图9。

图中各步骤的内容为：

1）SW_2 的FRACE功能模块检测到在端口2处发生了拥塞。

2）FRACE模块从流量控制配置文件中读取流控配置参数，该文件中含有网络管理员配置的限制发往目的主机 D_2 速率为10 Mbps的参数。

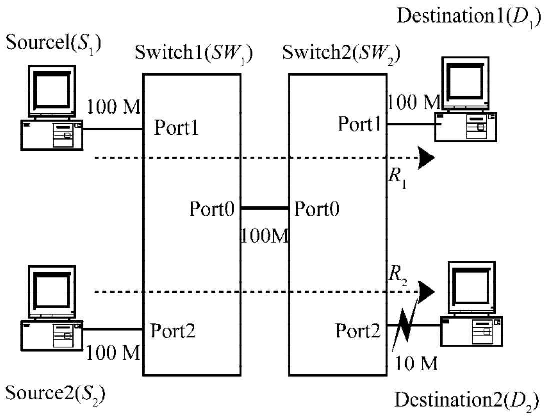

图8　速率失配网络图

Fig. 8　Rate mismatching network

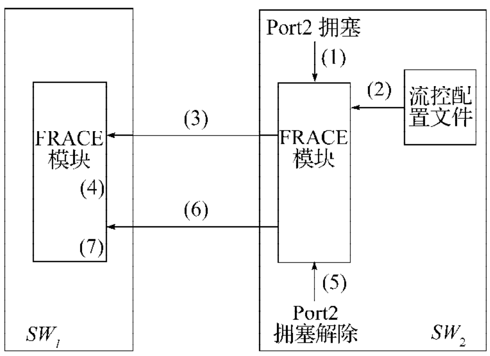

图9　FRACE流量控制过程示例

Fig. 9　FRACE flow control process sample

3）FRACE模块根据流控配置参数构造FRACE MAC控制帧，包含目的MAC地址为主机 D_2 的MAC

地址，控制速率为10 Mbps。并将该控制帧发往SW_1。

4）SW_1在收到该控制帧后，根据参数使能FRACE功能模块，并配置分类器和令牌桶，对所有目的地址为D_2的数据帧进行流控。

5）SW_2的FRACE功能模块检测到端口2处的拥塞解除。

6）FRACE模块构造FRACE MAC控制帧，包含目的MAC地址为主机D_2的MAC地址，控制速率为0xFFFFFFFE，指示取消该目的地址的流量控制。并将该控制帧发往SW_1。

7）SW_1在收到该控制帧后，禁止FRACE功能模块，解除流量控制。

3.4 FRACE流控功能验证

为了验证FRACE流控的有效性，我们使用C++语言建立了交换机模型和发送、接收数据的节点模型。在交换机模型中实现了FRACE流控能力，发送数据的节点的流量模型采用泊松模型。我们构造了如下两种典型的网络环境。

I **速率失配** 仿真网络模型如图10所示。

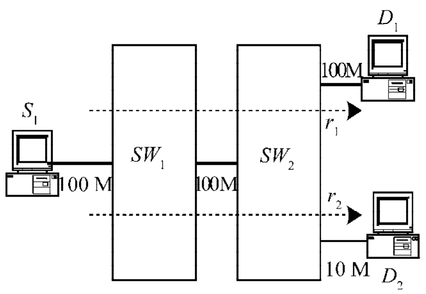

图10 速率失配仿真网络

Fig. 10 Rate mismatching emulation network

图中S_1为数据发送节点，其中的一部分数据以速率r_1发往目的节点D_1，另一部分以速率r_2发往目的节点D_2。我们设置速率r_2大于位于SW_2和D_2之间的链路带宽10 Mbps，以使得SW_2连接D_2的端口发生拥塞。我们分别配置SW_1和SW_2具有PAUSE或FRACE的流控能力，得到S_1到D_1和D_2数据流的仿真结果如图11和12所示。

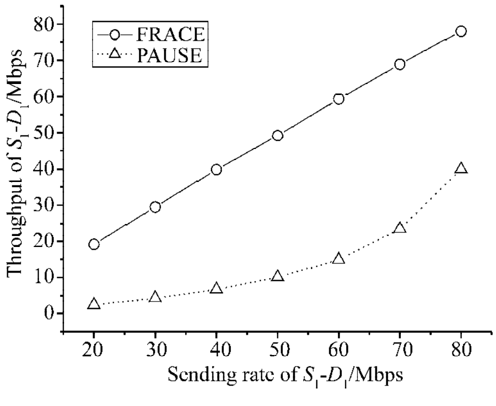

图11 S_1–D_1仿真结果

Fig. 11 S_1–D_1 emulation result

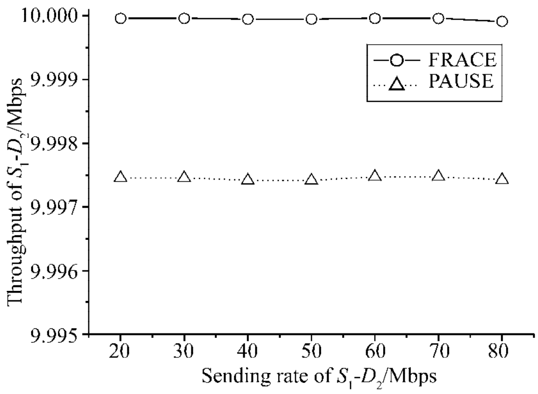

图12 S_1–D_2仿真结果

Fig. 12 S_1–D_2 emulation result

从图11、12中我们可以看到：

1）S_1–D_1：在PAUSE流控情况下，虽然SW_2到D_1之间的链路带宽有100 Mbps，大于S_1到D_1的数据发送速率，不会产生拥塞。但是由于S_1–D_2路径上产生的拥塞，使得PAUSE流控对S_1发往D_1的数据帧仍然进行了流量控制，其结果是S_1到D_1的实际数据通过量不到发送速率的一半。而在FRACE流控的情况下，我们在SW_2设置一个配置文件，配置参数使得S_1–D_2的数据流控制速率为10 Mbps。当SW_2发生拥塞时，根据配置文件产生控制S_1–D_2速率的FRACE控制帧发往SW_1，SW_1进行相应的速率控制操作，对S_1到D_1的数据流没有任何影响，其结果是S_1到D_1的数据通过量与发送速率一致。

2）S_1–D_2：在FRACE流控情况下，S_1到D_1的数据通过量基本维持在10 Mbps，比PAUSE流控更加精确（10 Mbps vs. 9.997 5 Mbps）。

从速率失配的仿真结果我们可以得出结论，FRACE流控能够实现精确的流量控制，并能避免不必要的流量控制，提高网络效率。

II **带宽公平** 仿真网络模型如图13所示。

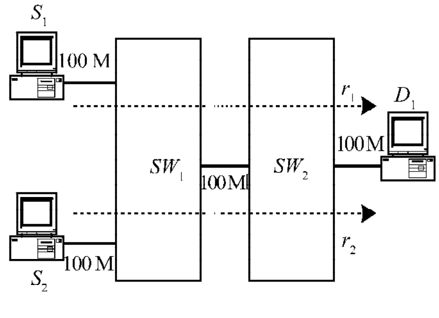

图13 带宽公平仿真网络

Fig. 13 Bandwidth equitable emulation network

图中S_1和S_2为数据发送节点，均为服务提供商的同一等级的用户，且都签订了最小保障带宽为5 Mbps

的服务协约。S_1以8 Mbps的速率向目的节点D_1发送数据，同时S_2以可变速率向D_1发送数据，其变化范围为25 Mbps至85 Mbps。我们分别配置SW_1和SW_2具有PAUSE或FRACE的流控能力，得到S_1和S_2和D_1数据流的仿真结果如图14和15所示。

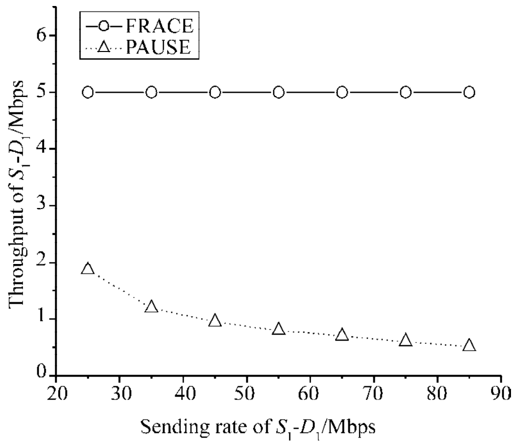

图14　S_1–D_1仿真结果

Fig. 14　S_1–D_1 emulation result

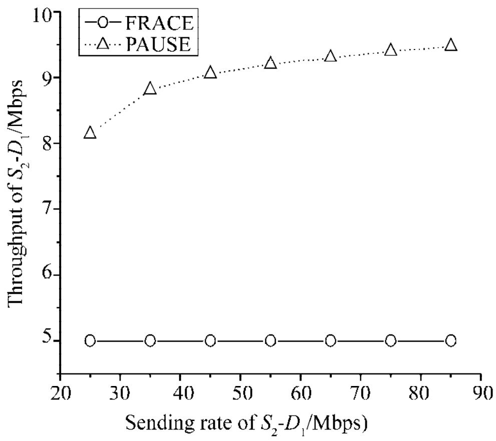

图15　S_2–D_1仿真结果

Fig. 15　S_2–D_1 emulation result

从图中我们可以看到：

1）在PAUSE流控的情况下，SW_2到D_1之间的10 Mbps链路带宽超过80 %被S_2占用，S_1发送数据的实际通过量基本维持在1 Mbps左右，远远小于服务提供商承诺的5 Mbps的最小带宽。造成这种情况的原因是S_1的数据发送速率过大，然而这并不是服务提供商能够控制的。

2）在交换机采用了FRACE流控后，我们在交换机中配置S_1和S_2的控制速率均为5 Mbps，从仿真结果图中我们可以看到，S_1和S_2到D_1的数据通过量均维持在5 Mbps，带宽公平性得到了很好的保证。

参考文献：

[1] ISO/IEC Standard 8802-3-1996，Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications [ISO publication of IEEE standard 802.3][S].

[2] Tanenbaum， Andrew S. Computer Networks[M]. [s.l.]：Prentice Hall，1988.

[3] 郑明春，杨寿保，于晓梅，孙伟峰. 一种提高异构网络传输性能的双向流量控制机制[J]. 电子学报，2006，34(5): 957-961.

[4] IEEE 802.3x-1998，Specification for 802.3 Full Duplex Operation. IEEE Standard 802.3[S].

[5] IEEE 802.1Q-1998，IEEE Standards for Local and Metropolitan Area Networks: Virtual Bridge Local Area Networks[S].

[6] 杨琳苹. 基于仿真的TCP流量控制机制的研究[J]. 四川理工学院学报：自然科学版，2006，19(3):75-80.