Implementing fair bandwidth allocation schemes in hose-modelled VPN

D. Wei and N. Ansari

Abstract: The virtual private network (VPN) provides customers with predictable and secure network connections over a shared network infrastructure. The recently proposed hose model for VPNs has desirable properties in terms of greater flexibility and better multiplexing gain. However, the 'classic' fair bandwidth allocation scheme introduces the issue of low overall utilisation in this model; furthermore, when the VPN links are established, the VPN customers cannot manage their VPN resources by themselves dynamically. The authors propose a fluid hose-modelled VPN, and based on this model they develop an idealised fluid fair bandwidth allocation scheme to improve the performance of the VPN. With the proposed scheme, they achieve two goals: maximising the overall throughput of the VPN; and providing a mechanism that enables the VPN customers to allocate the bandwidth according to their own requirements, thus achieving the predictable QoS performance. Based on deficit round robin (DRR), a transmission scheduling scheme for output buffer switches, a novel scheme, two-dimensional deficit round robin (2-D DRR), is developed to approximate/realise the idealised fluid fair bandwidth allocation scheme for the hose-modelled VPN. The simulation results also show that the 2-D DRR can improve the overall throughput without compromising fairness and implementation complexity.

1 Introduction

Virtual private network (VPN) refers to the communication between a set of sites, making use of a shared network infrastructure. Multiple sites of a private network may therefore communicate via the public infrastructure to facilitate the operation of the private network. The logical structure of the VPN, such as topology, addressing, connectivity, reachability and access control, is equivalent to part of or all of a conventional private network using private facilities.

Traditional VPNs are built based on layer 2 techniques (the pipe model in [1]), such as ATM [2] and frame relay [3]. Since resources, such as bandwidth and buffer, are explicitly partitioned, quality of service (QoS) can be guaranteed. Nowadays, the layer 3 VPN (IP-based VPN) is becoming prevalent owing to its desirable flexibility and scalability. However, the current layer 3 VPN is built based on a tunnelling technique over the Internet; since the Internet service is 'best-effort' in nature, it cannot provide guaranteed QoS. With emerging technologies, such as MPLS [4], RSVP [5], etc., it is possible to provide QoS in the layer 3 VPN. The hose-modelled VPN proposed in [1] has greater flexibility and better multiplexing gain within the customers of the same VPN.

In the pipe model, since the VPN customers have to specify QoS requirements between each pair of endpoints in the VPN, each VPN customer is required to know the complete traffic matrix, i.e. the traffic load between each pair of endpoints. There are two shortcomings in the pipe model: (i) since the number of endpoints of each VPN is increasing and the traffic pattern between endpoints becomes more complicated, it is difficult, for the VPN customers, to provide the traffic characteristics between pairs of endpoints; (ii) since each pipe (endpoint-to-endpoint) is created independently (such as in ATM, logically, it is a mesh topology), it is difficult to share the unused bandwidth among other pipes in the same VPN, i.e. no multiplexing gain among the customers in the same VPN.

In the hose model, the above-mentioned shortcomings can be alleviated. It is assumed that when a VPN is established, the architecture of the VPN and bandwidth reserved on each link between any two adjacent nodes are not changed. The VPN customer can specify QoS requirements per VPN endpoint instead of each pair of endpoints. As mentioned in [1], the hose model provides customers with the following advantages over the pipe model: (i) ease of specification of QoS – only one ingress and egress bandwidth per hose endpoint needs to be specified, as compared to the bandwidth for each pipe between pairs of hose endpoints; (ii) flexibility – traffic load from and to a hose endpoint can be distributed arbitrarily over other endpoints if the bandwidth of each endpoint is not violated; (iii) multiplexing gain among the same VPN users; and (iv) characterisation – hose requirements are easier to characterise because the statistical variability in the individual source-destination traffic is smoothed by aggregation into hoses.

However, in [1], the bandwidth in the 'hose' is partitioned according to traffic measurement and prediction. As we know, no traffic prediction scheme is perfect because the Internet traffic is uncertain and bursty in nature. Thus, we believe that, with a proper approach, the multiplexing gain could still be improved, and further greater overall throughput could be obtained. Furthermore, since the

IEE Proceedings online no. 20040840

doi:10.1049/ip-com:20040840

Paper first received 19th August 2003 and in revised form 1st June 2004. Originally published online: 5th November 2004

The authors are with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, University Heights, Newark NJ 07102, USA

bandwidth is reallocated according to traffic measurement and prediction, the VPN customers are not able to manage the VPN according to their requirements, and thus the predictable QoS performance cannot be achieved.

In this paper, based on the hose model, we propose a modified fluid VPN service model. With the newly proposed model, we develop a fluid fair bandwidth allocation scheme. As compared to the approach in the original hose model [1], we have two improvements: (i) maximise the overall throughput of VPN; and (ii) provide a mechanism that enables the VPN customers to manage the reserved bandwidth according to their requirements.

Deficit round robin (DRR) [6] is a transmission scheduling scheme for output buffer switches. Based on the proposed fluid model and DRR, we develop a new scheme, two-dimensional deficit round robin (2-D DRR), to approximate the idealised fluid fair bandwidth allocation scheme. By simulation results, we show that 2-D DRR can improve the overall throughput without compromising fairness and implementation complexity.

2 Background

In this Section, we review the hose-modelled VPN and the generic fluid model for fair bandwidth allocation, and point out the low utilisation issue, when the generic fluid bandwidth allocation scheme or max–min schemes are employed in the hose-modelled VPN.

2.1 Hose-modelled VPN

In the hose-modelled VPN, from the customer perspective, the VPN specification includes: (i) the set of VPN endpoints $P \subseteq V$, where V is the set of nodes in the network; (ii) for any node $i \in P$, the link capacity of the hose ingress and egress are L_i^{in} and L_i^{out}, respectively. From the service provider perspective, the VPN involves: (i) a tree T in topology due to its scalability, routing and restoration simplicity; (ii) for the link between node u and v in T, denote $T_{u,v}^u$ and $T_{u,v}^v$ as the connected components of T containing u and v, respectively, when the link between node u and v is deleted. Denote by $L_{u,v}$ the link capacity of the directional link from node u to v. Further denote $P_{u,v}^u$ and $P_{u,v}^v$ as the endpoint sets in $T_{u,v}^u$ and $T_{u,v}^v$, respectively.

$$L_{u,v} = \min\left(\sum_{\forall i \in P_{u,v}^u} L_i^{Out}, \sum_{\forall j \in P_{u,v}^v} L_j^{In} \right)$$

If $\forall i \in P$, $L_i^{in} = L_i^{out}$, the VPN tree T is called symmetric; otherwise, asymmetric. The notations adopted in this paper are described in Table 1.

Table 1: Notation used in this paper

V	set of nodes in VPN
E	set of bi-directional links between the nodes
P	set of nodes $P \subseteq V$ corresponding to VPN endpoints
T	tree of VPN
U	overall VPN throughput
i, j	endpoints in VPN, $i, j \in P$
u, v	nodes in VPN, $u, v \in V$
L_i^{In}	ingress link capacity for node i
L_i^{Out}	egress link capacity for node i
$f_j^{u,v}$	fair share rate of flows destined to endpoint j at the directional link from u to v
$a_{i,j}^u$	arrival rate of traffic from i to j at node u
$\phi_{i,j}$	weight of flow from node i to j

2.2 Idealised fluid fair bandwidth allocation scheme

The notations in [7] have been modified to make them consistent with ours. The fluid model for fair bandwidth allocation for the directional link from u to v, in [7], can be written as

$$\forall i \in P_{u,v}^u, j \in P_{u,v}^v, \; a_{i,j}^v = b_{i,j}^u = \min(a_{i,j}^u, f_{u,v}) \qquad (1)$$

where $f^{u,v}$ is the fair share of bandwidth for the directional link from u to v.

If $\sum_{\forall i \in P_{u,v}^u} \sum_{j \in P_{u,v}^v} a_{i,j}^u \le L_{u,v}$, then all traffic can be forwarded, and by convention, let

$$f^{u,v} = \max_{\forall i \in P_{u,v}^u, j \in P_{u,v}^v} a_{i,j}^u \qquad (2)$$

Otherwise, $f^{u,v}$ has to be computed iteratively as follows:

Step 1. Compute $f^{u,v} = L_{u,v}/N_{u,v}$, where $N_{u,v}$ is the number of flows on this link.

Step 2. Find the flow with the minimum allocated bandwidth.

Step 3. Subtract this rate in the link and eliminate the corresponding flow.

Step 4. Compute $f^{u,v}$ in the reduced set of flows as

$$f^{u,v} = \frac{L_{u,v} - \sum \text{rates of eliminated flows}}{N_{u,v} - \sum \text{number of eliminated flows}}$$

Step 5. Repeat steps 2–4 until all flows are eliminated.

2.3 Low utilisation issue

By means of the following example, we will demonstrate the problem of low utilisation if we employ (1) directly in the hose-modelled VPN. Note that in this paper one flow represents the traffic load starting from the same endpoint and ending at the other same endpoint.

Figure 1 shows a symmetric VPN, in which each endpoint has the hose link capacity of 2Mbit/s. H_0, H_1, H_2, H_3 and H_4 start to transmit packets to H_5 at time 0, 2, 4, 6, 8, at a constant rate of 0.2, 1, 1, 1, and 1 Mbit/s,

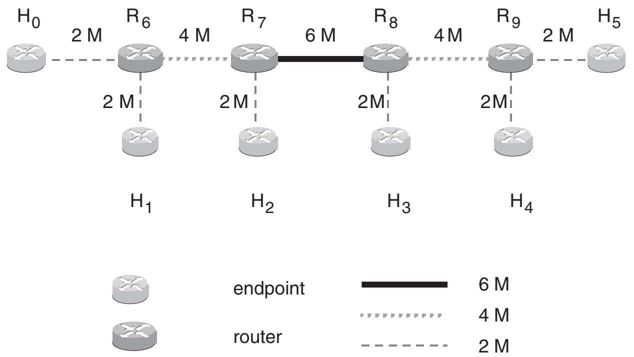

Fig. 1 *Hose-modelled VPN*

respectively (i.e., flow 0, 1, 2, 3, 4, respectively). H_0, H_1, H_2 and H_3 start to transmit packets to H_4 at time 0, 2, 4, 6, with load 0.2, 1, 0.4 and 0.4 Mbit/s, respectively (i.e. flow 5, 6, 7, 8, respectively). All flows are served equally. Thus, the bandwidth allocated to each flow can be computed by (1). NS-2 [8] is a network simulator widely used within the networking research community. It provides substantial support for conducting simulation of TCP, routing, scheduling, multicast protocols, etc. over wired and wireless (terrestrial and satellite) networks. We implement the deficit round robin (DRR) scheme [6] by NS-2 to conduct this experiment. Although DRR is not the best scheme to approximate the fair bandwidth scheme in the fluid model, it is desirable owing to its implementation simplicity. In this simulation, all traffics are UDP packets, with a fixed packet size of 500 bytes, and the quantum assigned to each flow is 500 bytes. The simulation results are shown in Figs. 2 and 3. From Fig. 3, we see that, at H_4, the totally received traffic, from time 6, cannot achieve 2 Mbit/s, although the link capacity of H_4 is 2 Mbit/s and we have enough traffic to H_4 (from time 6, the total traffic to H_4 is 2 Mbit/s). This is attributed to the following: at link R_8–R_9, in the time interval (6, 8), there are eight active flows (flows 0, 1, 2, 3, and 5, 6, 7, 8), the fair share of each flow is $(4-0.2-0.2-0.4-0.4)/4 = 0.7$ Mbit/s and so flow 1, 2, 3 and 5 should receive 0.7 Mbit/s bandwidth. However, since

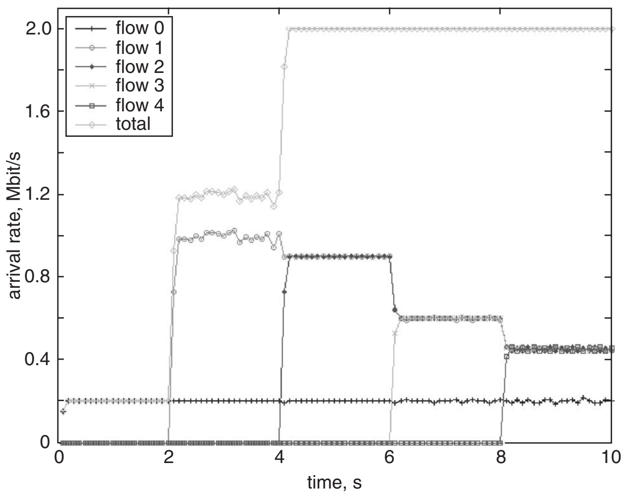

Fig. 2 *Arrival rates of flows at H_5*

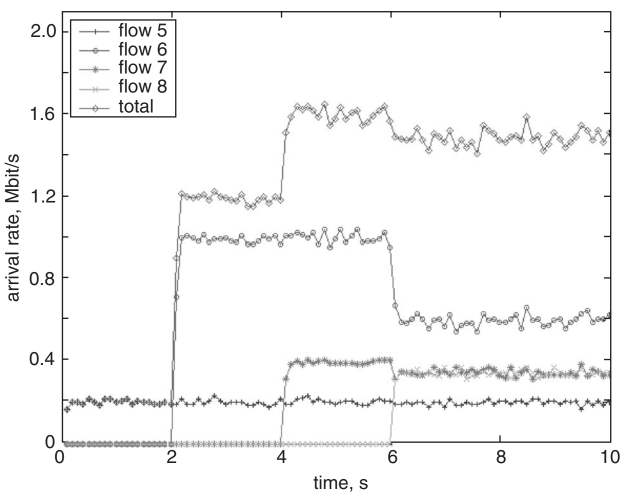

Fig. 3 *Arrival rates of flows at H_4*

flows 0, 1, 2, 3 are destined to H_5, their total traffic is 2.3 Mbit/s in link R_8–R_9, resulting in a drop of the excessive 0.3 Mbit/s traffic in the link R_8–R_9 because the link to H_5 is 2 Mbit/s. At the same time, the totally allocated bandwidth destined to H_4 is only $0.2+0.7+0.4+0.4 = 1.7$ Mbit/s, and again we have enough traffic (2 Mbit/s) to H_4, and the link from R_9 to H_4 has enough link capacity (2 Mbit/s). This happens because the fair bandwidth allocation of a single link could adversely affect the overall throughput. In this simulation, the fair bandwidth allocation at the link R_8–R_9 decreases the throughput of the link from R_9 to H_4.

Another approach is the max–min bandwidth reallocation scheme with consideration of bottleneck on other links, as described in [9]. The fair share rate can be computed as in Section 2.2, but in step 2 the minimum allocated bandwidth could be the one bottlenecked at the downstream links. This approach requires a feedback mechanism to inform a switch that, if any flow is bottlenecked at its downstream links, at moment 6 s, for example, R_9 should inform other routers that flow 1, 2 and 3 should be constrained, thus saving bandwidth for flow 6. This mechanism requires, at least, traffic measurement and a signalling protocol in order to dynamically reallocate the bandwidth. As we know, more accurate bandwidth reservation needs more accurate traffic measurement and more signalling overhead, thus consuming CPU processing power and more bandwidth, and further reducing the overall throughput of the VPN.

3 Proposed fluid VPN model

As discussed in Section 2, there are several shortcomings if current approaches are employed directly in the hose-modelled VPN. We develop a new fluid model based on the original hose model. In the new model, an idealised fair bandwidth allocation scheme is developed by taking the link capacity of endpoints into consideration. It is proved theoretically that the proposed scheme maximises the overall throughput without compromising its fairness property.

A VPN is said to be a fluid hose-modelled VPN if it possesses the following properties:

Property 1. This VPN has a tree topology.

Property 2. If nodes u and v are linked,

$$L_{u,v} = \min\left(\sum_{\forall i \in P^u_{u,v}} L^{Out}_i, \sum_{\forall j \in P^v_{u,v}} L^{In}_i\right)$$

Property 3. The bandwidth of the link to endpoint j is fairly allocated among the flows destined to j, by using

$$a^j_{i,j} = \min(f_j \cdot \phi_{i,j}, a^i_{i,j}) \quad (3)$$

where $\sum_{\forall i \in P} \phi_{i,j} = 1$, and f_j, the fair share rate of the link leading to j, is computed the same way as in Section 2, as if all other endpoints are connected to the same link leading to j.

Note that the first two properties are the same as those in the original hose model. With the third property, the VPN customers are able to allocate bandwidth, at the endpoint, by setting the weights, $\phi_{i,j}$, according to their requirements.

Lemma 1: In the proposed fluid VPN model, $\forall j \in P$,

$$\sum_{\forall i \in P} a^j_{i,j} \le \min\left(\sum_{\forall i \in P} a^i_{i,j}, L^{Out}_j\right) \quad (4)$$

Proof: For any flow starting from node i and ending at node j, packets generated at node i could be discarded at the intermediate nodes; then $a^j_{i,j} \le a^i_{i,j}$. Thus, $\sum_{\forall i \in P} a^j_{i,j} \le \sum_{\forall i \in P} a^i_{i,j}$.

The aggregated traffic destined to j is limited by the egress link capacity of the link leading to j, L^{Out}_j, and hence

$$\sum_{\forall i \in P} a^j_{i,j} \le L^{Out}_j$$

Therefore,

$$\sum_{\forall i \in P} a^j_{i,j} \le \min\left(\sum_{\forall i \in P} a^i_{i,j}, L^{Out}_j\right) \qquad \square$$

Lemma 2: In the proposed fluid VPN model, the overall VPN throughput

$$U \le \sum_{\forall j \in P}\left(\min\left(\sum_{\forall i \in P} a^i_{i,j}, L^{Out}_j\right)\right) \tag{5}$$

Proof: The overall VPN throughput

$$U = \sum_{\forall j \in P}\sum_{\forall i \in P} a^j_{i,j}.$$

By inequality (4),

$$U = \sum_{\forall j \in P}\sum_{\forall i \in P} a^j_{i,j} \le \sum_{\forall j \in P}\left(\min\left(\sum_{\forall i \in P} a^i_{i,j}, L^{Out}_j\right)\right) \qquad \square$$

Lemma 1 tells us that the maximum throughput of an egress link to endpoint j is $\min(\sum_{\forall i \in P} a^i_{i,j}, L^{Out}_j)$. Lemma 2 demonstrates that the overall throughput is maximised when the throughput of each egress link is maximised.

Theorem 1: In any directional link from u to v, if

$$\forall i \in P^u_{u,v}, j \in P^v_{u,v}, a^v_{i,j} = \min\left(a^u_{i,j}, f^{u,v}_j \cdot \frac{\phi_{i,j}}{\sum_{k \in P^u_{u,v}} \phi_{k,j}}\right) \tag{6}$$

where the fair share rate $f^{u,v}_j$ can be computed the same way as in Section 2.2, but as if the link capacity for flows destined to endpoint j on the directional link from u to v is the same as the egress link to endpoint j, i.e. the total available bandwidth, on the link from u to v, for flows destined to endpoint j is L^{Out}_j. Then: (i) property 3 of the definition of the fluid hose-modelled VPN can be met; (ii) the maximum overall VPN throughput can be achieved; (iii) this scheme can be employed without violating the link capacity of each link, i.e.

$$\sum_{\forall j \in P^v_{u,v}}\sum_{\forall i \in P^u_{u,v}} a^v_{i,j} \le L_{u,v}$$

The detailed proof of theorem 1 can be found in the Appendix. Note that the difference between our fluid model and the original fluid model for fair bandwidth allocation is that we use one fair share rate for flows destined to the same endpoint whereas, in the original fluid model, a global fair share rate is employed at a single link.

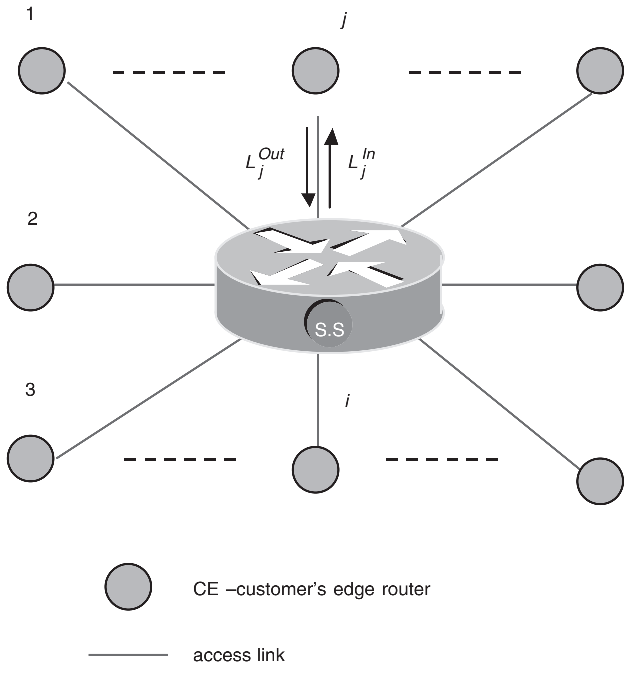

Fig. 4 *Simplified 'superswitch' model*

With property 3 of the fluid hose-modelled VPN, from the VPN customer perspective, the set of all intermediate nodes and the links among them can be considered as a simplified 'superswitch' model, as shown in Fig. 4. At endpoint j, the VPN customer is able to allocate the bandwidth of the link leading to j, by assigning weight $\phi_{i,j}$ to the flow from i to j, $\forall i \in P$; then $a^j_{i,j}$, the bandwidth allocated to this flow, can be guaranteed no less than $\phi_{i,j} \cdot L^{Out}_j$, when there is enough traffic in this flow. Therefore, the proposed, idealised fluid bandwidth allocation scheme possesses the following desirable properties: (i) the VPN customers obtain a predictable QoS in terms of guaranteed bandwidth; (ii) the maximum throughput of the VPN; (iii) the fairness among all flows destined to the same destination; and (iv) a mechanism that enables the VPN customers to manage their VPN resource in terms of bandwidth according to their own requirements.

4 Implementation

The proposed fluid model is an idealised model, which cannot be implemented in the real world. We develop a modified deficit round robin scheme, called the two-

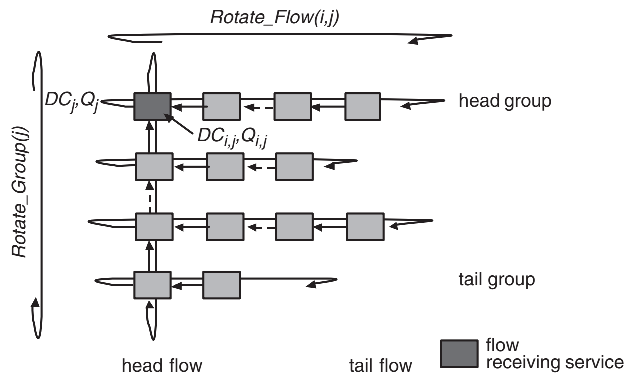

Fig. 5 *Architecture of two-dimensional deficit round robin*

```
Initialisation procedure :
for ( j=0; j<|D|; j++)
    DC_j=0;
    for ( i=0; i <|S|; i++)
        DC_i,j=0;

Enqueueing procedure: //on arrival of packet p
Session(i,j) = ExtractFlow( p)
if (ExistsInActiveGroupList( j)==FALSE) then
    AppendToActiveGroupList( j);
    DC_j=0;
if (ExistsInActiveGroupList(i,j)==FALSE) then
    AppendToActiveGroupList(i,j);
    DC_i,j=0;
if (no_free_buffer_left==TRUE) then
    Discard_Packet( p);
else
    Enqueue( p, Flow(i,j));

Dequeueing procedure:
While (ActiveListIsEmpty==FALSE)
    While (DC_j !=0)
        if (ExistsInActiveGroupList(i,j)==TRUE) then
            if (DC_j>= Q_i,j)
                DC_i,j= DC_i,j+Q_i,j;
                DC_j= DC_j-Q_i,j;
            if (p_i,j^Head<= Q_i,j) then
                Send(p_i,j^Head);
                DC_i,j= DC_i,j-PacketSize(p_i,j^Head);
                Rotate_Flow(i,j);
            else
                DC_i,j = DC_i,j+DC_j;
                DC_j = 0;
        else
            DC_j = 0;
    Rotate_Group( j);
    DC_j= Q_j;
```

Fig. 6 *Two-dimensional deficit round robin scheme*

dimensional deficit round robin (2-D DRR), to approximate the scheme in the proposed fluid model. This scheme maintains the same desirable implementation simplicity as DRR. Figs. 5 and 6 show the architecture and pseudo-code of 2-D DRR, respectively.

The principle of the proposed scheme is that the unused bandwidth of one flow is shared among those flows destined to the same endpoint, and thus the throughput of the egress link to this destination can be maximised. Considering the constraint of the link capacity of all endpoints, we place flows into different groups according to their destined endpoints, i.e. flows with the same destination address are placed into the same group. Each group has its own deficit counter and a group quantum. Note that the quantum of each flow is proportional to the weight $\phi_{i,j}$ in the idealised fluid hose-modelled VPN.

We know that the allocated bandwidth to a flow is proportional to its assigned quantum, and thus

$$\forall i,j,k,l \in P, \quad \frac{Q_{i,j}}{Q_{k,l}} = \frac{\phi_{i,j} L_j^{Out}}{\phi_{k,l} L_l^{Out}} \tag{7}$$

To maintain the scheduling implementation simplicity of $O(1)$, each backlogged flow must be served whenever visited by the scheduler. This property can be guaranteed by the following inequality (7):

$$Q_{\min} = \min_{\forall i,j \in P} (Q_{i,j}) \geq \max(PacketSize) \tag{8}$$

The minimum queueing delay is also desirable, and we thus need to minimise the time interval to visit a flow. Therefore, let

$$Q_{\min} = \min_{\forall i,j \in P} (Q_{i,j}) = \max(PacketSize) \tag{9}$$

Then,

$$\forall j \in P, \quad Q_j = \sum_{\forall i \in P} Q_{i,j} \tag{10}$$

By (7), we first find the flow which demands the smallest bandwidth, and assign to this flow the quantum, $Q_{\min}$, to be the maximum packet size according to (9). The quantum assigned to each flow, $Q_{i,j}$, can be computed by (7). Finally, by (10), the group quantum Q_j can be computed.

The proposed scheme can be summarised as follows:

- When the VPN is set up, the quantum of each flow, $Q_{i,j}$, and the quantum assigned to each group, Q_j, in each router, are computed as above. Each group deficit counter, DC_j, and flow deficit counter, $DC_{i,j}$, are set to 0.
- When a new packet arrives, the enqueueing procedure is initiated. First, it checks if the group to which this flow belongs, is in the active group list. If not, this group is appended to the tail of the group list. Second, it checks if the corresponding flow is in the active flow list. If not, this flow is appended to the active flow list. Finally, if there are enough buffers, this packet is placed in the corresponding queue; otherwise it is discarded.
- The dequeueing procedure always operates at the head flow (i.e. the first flow of the group) in the head group, e.g. flow (i, j), as shown in Fig. 4. When a group is rotated to the head, the group counter DC_j is set to the group quantum Q_j. First, if the group counter DC_j is not less than the flow quantum $Q_{i,j}$, the flow deficit counter $DC_{i,j}$ is increased by the flow quantum $Q_{i,j}$, and the group deficit counter DC_j is decreased by $Q_{i,j}$; otherwise, the $DC_{i,j}$ is increased by DC_j, and DC_j is set to 0. Second, if the size of the first packet in this flow, $PacketSize(p_{i,j}^{Head})$, is not greater than $DC_{i,j}$, this packet is transmitted and $DC_{i,j}$ is decreased by $PacketSize(p_{i,j}^{Head})$. Third, procedure *Rotate_Flow*(i,j) dequeues this flow and appends it to the tail of the group, and then places the next flow at the head. Finally, if the group deficit counter DC_j is not 0, the procedure goes back to the first step; otherwise, procedure *Rotate_Group*(j) takes this group from the head and appends it to the tail of the group list, and then places the next group at the head. Note that, when a flow or a group becomes idle, it is dequeued from the corresponding active list.

Clearly, every active flow is visited in a round robin fashion and each active flow must be served when it is visited. Therefore, the implementation complexity of the 2-D DRR scheme is still $O(1)$, the same as DRR.

5 Simulation results

We implement the 2-D DRR scheme for the VPN shown in Fig. 1 by NS-2 [8]. With the same traffic pattern from sources as described in Section 2, we define each flow with a weight proportional to its ingress link capacity, i.e. each flow has the same weight. The group quantum assigned to H_4 and H_5 are both 2500 bytes, and the quantum assigned to each flow is 500 bytes. The arrival traffic patterns of flows 0–4 at H_5 are shown in Fig. 7, and are almost the same as those in Fig. 2. The arrival traffic patterns of flows 5–8 at H_4 are shown in Fig. 8. Note that the arrival rates of flows 5, 7 and 8 are almost the same

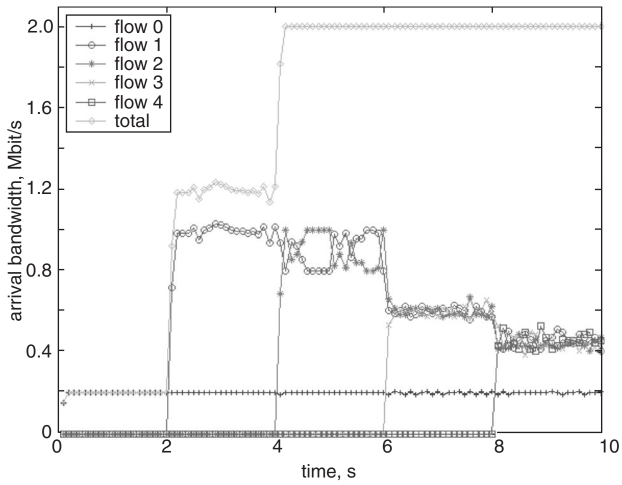

Fig. 7 *Arrival rates of flows at H_5 (with 2-D DRR)*

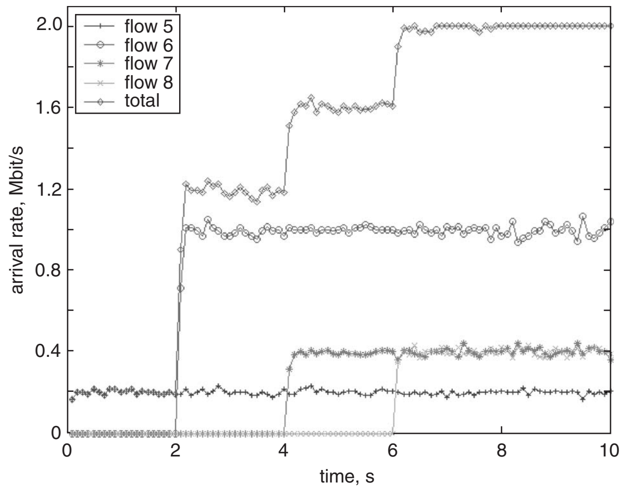

Fig. 8 *Arrival rates of flows at H_4 (with 2-D DRR)*

as in Fig. 3, whereas the arrival rate of flow 6 is 1 Mbit/s from time 2 s, which is what we want. Thus, the traffic throughput of link R_9–H_4 is improved from 1.7 Mbit/s to 2 Mbit/s, while the traffic throughput of link R_9-H_5 remains 2 Mbit/s, from time 6 s. Therefore, the overall VPN throughput is improved from 3.7 Mbit/s to 4.0 Mbit/s from time 6.

6 Discussions

As discussed in Section 1, in layer 2 VPN, since resources such as bandwidth and buffer are explicitly partitioned, quality of service (QoS) can be guaranteed, whereas the layer 3 VPN cannot provide guaranteed QoS, despite its desirable flexibility and scalability. The proposed bandwidth allocation scheme, instead of replacing the current VPN techniques, can be integrated within the current framework of layer 3 VPN. With the proposed scheme and RSVP [5], we can obtain the best of both layer 2 and 3 VPNs, i.e. service providers can provide guaranteed or predictable QoS, as in layer 2 VPNs, while maintaining the flexibility and simplicity of layer 3 VPN.

We have also conducted experiments with other traffic patterns, such as TCP, Pareto and Poisson; the results are similar to those presented in Section 5, and the same conclusions can thus be drawn therein. Owing to the limited space and to avoid being repetitive, these results are not presented here. Therefore, the proposed schemes can provide predictable QoS in terms of bandwidth, regardless of the traffic pattern. Although the deployment issue requires substantial investigation and coverage, we believe the proposed scheme can be readily deployed and integrated within the framework of the virtual-router-based VPN [10], implemented at all intermediate nodes in the ISP's network.

The proposed scheme can benefit not only the VPN customers, but also service providers. It is clear that, if a packet has to be discarded before it arrives its destination, it should be discarded as early as possible, thus avoiding congestion and resource wastage in the network. In the proposed fluid bandwidth allocation scheme, knowing the egress link capacity of an endpoint, the intermediate routers are able to drop those packets at the earliest stage, without the knowledge of the downstream traffic pattern. Since ISPs may carry many customers' traffic on a single link, then if some bandwidth from a VPN customer is saved, it can be used to carry best-effort traffic of other customers, because the VPN customers would observe the same result even if those packets are not discarded earlier. In this case, the ISP is able to 'steal' some bandwidth from the VPN link without noticeable performance degradation from the VPN customers' perspective. In this sense, the overall throughput of the ISP's network can be improved with the proposed scheme. Furthermore, with the development and growth of the Internet, Internet service providers (ISPs) are required to offer revenue-generating and value-added services instead of only providing bandwidth and access services. VPN is one of the most important value-added services for ISPs. According to [11], an ideal provider-provisioned VPN should provide a mechanism that enables the VPN customers to manage their own VPN resource according to their own requirements. The proposed scheme does facilitate this manageability. By assigning different weights to different flows, the bandwidth to the same destination can be allocated to different flows according to the requirements of VPN users. The desirable properties of the proposed scheme can therefore enhance the ISP's competitive power.

7 Conclusions

In this paper, we have presented a fluid VPN service model which provides a mechanism that enables the VPN customers to allocate the egress link bandwidth arbitrarily according to their requirements. We have also proposed an idealised fluid fair bandwidth allocation scheme in this model. It is shown, analytically, that this scheme is able to maximise the overall VPN throughput. We have further developed a practical scheme, which has the desirable of simplicity implementation, to approximate the idealised fluid bandwidth allocation scheme. Simulation results have demonstrated that the proposed 2-D DRR algorithm can meet our objectives. As compared to the original measuring–predicting–reserving mechanism in [1], the proposed scheme maximises the overall VPN throughput without the overhead for traffic measurement, prediction, and signalling.

8 Acknowledgment

This work was supported in part by the NJ Commission on Science and Technology via NJWINS, and the New Jersey Commission on Higher Education via the NJ I-TOWER project.

9 References

1 Duffield, N.G., Goyal, P., Greenberg, A., Mishra, P., Ramakrishnan, K.K., and van der Merwe, J.E.: 'Resource management with hoses: point-to-cloud services for virtual private networks', *IEEE/ACM Trans. Netw.*, 2002, **10**, (5), pp. 679–692
2 Black, U.: 'ATM: Foundation for broadband networks' (Prentice Hall PTR, 1997)
3 Buckwalter, J.: 'Frame relay: technology and practice' (Addison-Wesley, 1999)
4 Rosen E., Viswanathan A., and Callon R.: 'Multiprotocol label switching architecture', IETF RFC 3031, January 2001, http://www.ietf.org/rfc/rfc3031.txt
5 Braden R., Zhang L., Berson S., Herzog S., and Jamin S.: 'Resource ReSerVation Protocol (RSVP) – Version 1 functional specification', IETF RFC 2205, http://www.ietf.org/rfc/rfc2205.txt, September 1997
6 Shreedhar, M., and Varghese, G.: 'Efficient fair queueing using deficit round robin', *IEEE/ACM Trans. Netw.*, 1996, **4**, (3), pp. 375–385
7 Stoica, I., Shenker, S., and Zhang, H.: 'Core-stateless fair queueing: achieving approximately fair bandwidth allocations in high speed networks'. Proc. SIGCOM '98, Vancouver, BC, Canada, 1998, pp. 118–130
8 'The Network Simulator – NS-2', http://www/isi.edu/nsnam/ns, May 2004
9 Arulambalam, A., Chen, X., and Ansari, N.: 'Allocating fair rates for available bit rate service in ATM networks', *IEEE Commun. Mag.*, 1996, **34**, (11), pp. 92–100
10 Knight P., Ould-Brahim H., Wright G., Gleeson B., Sloane T., Bubenik R., Sargor C., Negusse I., and Yu J.: 'Network based IP VPN architecture using virtual routers'. Internet draft, IETF. April 2004. http://www.ietf.org/internet-drafts/draft-ieft-13vpn-vpn-vr-02.txt
11 Chiussi F., De Clercq J., Ganti S., Lau W., Nandy B., Seddigh N., and Van den Bosch S.: 'Framework for QoS in provider-provisioned VPNs'. Internet draft, IETF. November, 2001. http://www.ietf.org/proceedings/01dec/I-D/draft-ietf-ppvpn-framework-02.txt

10 Appendix

To prove theorem 1, we need the following lemma.
Lemma 3: Suppose node m is the only intermediate node directly connected to endpoint j; if the bandwidth allocation scheme described in Section 3 is employed, then $a_{i,j}^{j} = \min(a_{i,j}^{i}, f_j^{m,j} \cdot \phi_{i,j})$.
Proof: Without loss of generality, we assume the flow starting from endpoint i and destined to endpoint j travels in the order of node 1, 2, …, m, and at the end arrives at endpoint j. Since traffic from i to j cannot be generated at any intermediate nodes, thus $a_{i,j}^{m} \le a_{i,j}^{m-1} \ldots\ldots \le a_{i,j}^{1} \le a_{i,j}^{i}$. If the proposed fair bandwidth allocation scheme is employed at each intermediate node, then

$$\forall l = 1,2,\ldots,m-1,m,$$

$$a_{i,j}^{l} = \min\left(a_{i,j}^{l-1}, f_j^{l-1,l}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l-1,l}^{l-1}}\phi_{k,j}}\right)$$

Thus,

$$\forall l = 1,2,\ldots,m-1,m,$$

$$a_{i,j}^{l+1} = \min\left(a_{i,j}^{l}, f_j^{l,l+1}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l,l+1}^{l}}\phi_{k,j}}\right)$$

$$= \min\left(a_{i,j}^{l-1}, f_j^{l-1,l}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l-1,l}^{l-1}}\phi_{k,j}}, f_j^{l,l+1}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l,l+1}^{l}}\phi_{k,j}}\right)$$

(i) If $a_{i,j}^{l-1} \le f_j^{l-1,l}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l-1,l}^{l-1}}\phi_{k,j}}$,

$$a_{i,j}^{l+1} = \min\left(a_{i,j}^{l-1}, f_j^{l,l+1}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l,l}^{l}+1}\phi_{k,j}}\right)$$

(ii) If $a_{i,j}^{l-1} > f_j^{l-1,l}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l-1,l}^{l-1}}\phi_{k,j}}$, then $\sum_{\forall k\in P_{l-1,l}^{l-1}} a_{i,j}^{l-1} > L_j^{Out}$

Since $P_{l-1,l}^{l-1} \subseteq P_{l,l+1}^{l}$, the same bandwidth of L_j^{Out} is shared on the link from l to $l+1$ by more flows than on the link from $l-1$ to l, and thus $f_j^{l,l+1} \le f_j^{l-1,l}$. Therefore,

$$f_j^{l,l+1}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l,l+1}^{l}}\phi_{k,j}} \le f_j^{l-1,l}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l-1,l}^{l-1}}\phi_{k,j}}$$

Then,

$$a_{i,j}^{l+1} = \min\left(a_{i,j}^{l-1}, f_j^{l,l+1}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l,l}^{l}+1}\phi_{k,j}}\right)$$

$$\therefore\ \forall l = 1,2,\ldots,m-1,m,$$

$$a_{i,j}^{l+1} = \min\left(a_{i,j}^{l-1}, f_j^{l,l+1}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{l,l+1}^{l}}\phi_{k,j}}\right)$$

Now for $i=0$, we can obtain

$$a_{i,j}^{j} = \min\left(a_{i,j}^{i}, f_j^{m,j}\frac{\phi_{i,j}}{\sum_{\forall k\in P_{m,j}^{m}}\phi_{k,j}}\right)$$

Since $P_{m,j}^{m} = P\backslash\{j\}$, and no traffic is allowed to both originate from and end at the same endpoint, then

$$\sum_{\forall k\in P_{m,j}^{m}}\phi_{k,j} = \sum_{\forall k\in P}\phi_{k,j} = 1$$

Therefore, $a_{i,j}^{j} = \min(a_{i,j}^{i}, f_j^{m,j}\cdot\phi_{i,j})$ □

Proof of Theorem 1:
(i) Again, without loss of generality, we assume the flow starting from endpoint i and destined to endpoint j travels in the order of node 1, 2,…, m, and at the end arrives endpoint j. Let P_1^j denote the set of endpoints, such that $\forall i \in P_1^j,\ a_{i,j}^{i} \le f_j^{m,j}\cdot\phi_{i,j}$, where m is the intermediate node which is connected to endpoint j; let P_2^j denote the set of endpoints, such that $\forall i \in P_2^j,\ a_{i,j}^{i} > f_j^{m,j}\cdot\phi_{i,j}$. Hence, $P = P_1^j \cup P_2^j$ and $P_1^j \cap P_2^j = \varnothing$.

Therefore,$\forall i \in P_1^j$, when the proposed scheme is employed on each node, by lemma 3, $a_{i,j}^{j} = a_{i,j}^{i}$. Then the fair share rate at the directional link

$$f_j^{m,j} = \begin{cases} \dfrac{L_j^{Out} - \sum_{\forall i\in P_1^j} a_{i,j}^{i}}{\sum_{\forall i\in P_2^j}\phi_{i,j}}, & \text{if } P_2^j \ne \varnothing \\ \max\limits_{\forall k\in P}\dfrac{a_{i,j}^{k}}{\phi_{k,j}}, & \text{if } P_2^j \ne \varnothing \end{cases} \quad (11)$$

$\forall i \in P_2^j$, by lemma 3, $a_{i,j}^{j} = f_j^{m}\cdot\phi_{i,j}$.

Now we need to prove that, in the simplified 'superswitch' model, the allocated bandwidth of each flow (i.e. property 3 of the definition of the fluid hose-modelled VPN) is the same as that we use in the proposed scheme, i.e. $\forall i \in P$, $a_{i,j}^{j'} = a_{i,j}^{j}$, where $a_{i,j}^{j'}$ is the allocated bandwidth in the simplified 'superswitch' model. We will prove this by contradiction.

Let $P_1^{j'}$ denote the set of endpoints, such that $\forall i \in P_2^{j'}$, $a_{i,j}^{i} \leq f_j$, where f_j is the fair share rate of endpoint j in the simplified 'superswitch' model; let $P_2^{j'}$ denote the set of endpoints, such that $\forall i \in P_2^{j'}$, $a_{i,j}^{i} > f_j$. Hence, $P = P_1^{j'} \cup P_2^{j'}$ and $P_1^{j'} \cap P_2^{j'} = \varnothing$;

$$f_j = \frac{L_j^{Out} - \sum_{\forall i \in P_1^{j'}} a_{i,j}^{i}}{\sum_{\forall i \in P_2^{j'}} \phi_{i,j}} \quad (12)$$

Assume $P_1^{j} \subset P_1^{j'}$. Then $P_2^{j'} \subset P_2^{j}$, and

$$\sum_{\forall i \in P_1^{j}} a_{i,j}^{i} < \sum_{\forall i \in P_1^{j'}} a_{i,j}^{i}, \sum_{\forall i \in P_2^{j'}} \phi_{i,j} < \sum_{\forall i \in P_2^{j}} \phi_{i,j}$$

By (11) and (12), $f_j^{m,j} < f_j$. Thus, $P_1^{j'} \subseteq P_1^{j}$, which is contradictory to the assumption. Therefore, $P_1^{j} \not\subset P_1^{j'}$.

Assume $P_1^{j'} \subset P_1^{j}$. Then $P_2^{j} \subset P_2^{j'}$, and

$$\sum_{\forall i \in P_1^{j'}} a_{i,j}^{i} < \sum_{\forall i \in P_1^{j}} a_{i,j}^{i}, \sum_{\forall i \in P_2^{j}} \phi_{i,j} < \sum_{\forall i \in P_2^{j'}} \phi_{i,j}$$

By (11) and (12), $f_j < f_j^{m,j}$. Thus, $P_1^{j} \subseteq P_1^{j'}$, which is contradictory to the assumption. Therefore, $P_1^{j'} \not\subset P_1^{j}$.

Hence, $P_1^{j'} = P_1^{j}$, $P_2^{j'} = P_2^{j}$ and $f_j = f_j^{m,j}$. Then, $\forall i \in P$, $a_{i,j}^{j} = a_{i,j}^{j'}$ and property 3 of the definition of the fluid hose-modelled VPN can be met.

(ii) By (3),

$$\sum_{\forall i \in P} a_{i,j}^{j} = \sum_{\forall i \in P} (\min(a_{i,j}^{i}, f_j \cdot \phi_{i,j}))$$

If $P_2^{j} = \varnothing$,

$$\sum_{\forall i \in P} a_{i,j}^{j} = \sum_{\forall i \in P} a_{i,j}^{i}$$

if $P_2^{j} \neq \varnothing$, by (11),

$$\sum_{\forall i \in P} a_{i,j}^{j} = L_j^{Out}.$$

$$\therefore \sum_{\forall i \in P} a_{i,j}^{j} = \min(\sum_{\forall i \in P} a_{i,j}^{i}, L_j^{Out})$$

By lemma 1, the throughput of the directional link leading to endpoint j is maximised when the proposed scheme is employed. Furthermore, by lemma 2, the overall VPN throughput is maximised when the throughput on the directional link leading to each endpoint is maximised.

(iii) By property 2 of the definition of the fluid hose-modelled VPN,

$$L_{u,v} = \min(\sum_{\forall i \in P_{u,v}^{u}} L_i^{Out}, \sum_{\forall j \in P_{u,v}^{v}} L_i^{In})$$

When the proposed scheme is employed, the flow destined to endpoint j is constrained, at each node, by the egress link capacity L_j^{Out}, i.e.

$$\sum_{\forall i \in P_{u,v}^{u}} a_{i,j}^{v} \leq L_j^{Out}$$

and thus

$$\sum_{\forall j \in P_{u,v}^{v}} \sum_{\forall i \in P_{u,v}^{u}} a_{i,j}^{v} \leq \sum_{\forall j \in P_{u,v}^{v}} L_j^{Out}$$

In the proposed fluid hose-modelled VPN, aggregate flows originated from endpoint i are also constrained by the ingress link capacity L_i^{Out}, i.e.

$$\sum_{\forall j \in P_{u,v}^{v}} a_{i,j}^{i} \leq L_i^{In}$$

$$\because a_{i,j}^{v} \leq a_{i,j}^{i}$$

$$\therefore \sum_{\forall j \in P_{u,v}^{v}} a_{i,j}^{v} \leq L_i^{In}.$$

Thus,

$$\sum_{\forall i \in P_{u,v}^{u}} \sum_{\forall j \in P_{u,v}^{v}} a_{i,j}^{v} \leq \sum_{\forall i \in P_{u,v}^{u}} L_i^{In}$$

Therefore,

$$\sum_{\forall j \in P_{u,v}^{v}} \sum_{\forall i \in P_{u,v}^{u}} a_{i,j}^{v} \leq \min(\sum_{\forall i \in P_{u,v}^{u}} L_i^{In}, \sum_{\forall j \in P_{u,v}^{v}} L_j^{Out}) = L_{u,v} \quad \square$$