On Dynamic Control Parameter Configuration Mechanism for Inter- and Intra-VPN Fairness Control Mechanism

† Osamu Honda †Hiroyuki Ohsaki †Makoto Imase ‡Junichi Murayama ‡Kazuhiro Matsuda

†Graduate School of Information Science and Technology, Osaka University, Japan
E-mail:{o-honda, oosaki, imase}@ist.osaka-u.ac.jp

‡NTT Information Sharing Platform Laboratories, NTT Corporation, Japan
E-mail:{murayama.junichi, matsuda.kazuhiro}@lab.ntt.co.jp

Abstract

In our previous work, we have proposed an IP-VPN fairness control mechanism called I2VFC (Inter-and Intra-VPN Fairness Control) that achieves fairness among IP-VPN customers. In order for I2VFC to achieve fairness among IP-VPN customers in arbitrary network configurations, I2VFC control parameters need to be automatically configured. In this paper, we first discuss design goals of a dynamic control parameter configuration mechanism for I2VFC. We then propose a dynamic control parameter configuration mechanism called DCPC (Dynamic Control Parameter Configuration) that automatically configures I2VFC control parameters by introducing the concept of a virtual VPN flow called *nominal VPN flow*. Through several simulation experiments, we quantitatively show how accurately and on what timescale inter-VPN fairness is realized using our dynamic control parameter configuration mechanism DCPC. Consequently, we show that I2VFC can achieve fairness with high accuracy in several network configurations using our dynamic control parameter configuration mechanism DCPC. We also show that I2VFC control parameters can follow network changes on the timescale of approximately 100 times of the round-trip time of VPN flows.

1 Introduction

IP-based virtual private networks (IP-VPNs), which provide a virtual privately owned network over an IP network, have attracted attention [1–3]. A virtual private network can be constructed on an IP network at a far lower cost than with conventional dedicated lines.

However, there is a serious problem that existing IP-VPNs cannot guarantee fairness among IP-VPN customers. This is because an IP network is a best-effort network, so that the IP-VPN constructed on it is also a best-effort network. In our previous work [4], we have proposed I2VFC (Inter- and Intra-VPN Fairness Control) to realize fair IP-VPN services within a layer 3 provider-provisioned VPN (L3-PPVPN) framework [5].

I2VFC is an AIMD (Additive Increase and Multiplicative Decrease) window flow control [6] that operates between IP-VPN service provider's edge routers (PE routers). In order to achieve fairness among IP-VPN customers in various network configurations, several I2VFC control parameters should be dynamically configured according to the status of a network. This paper first discusses the following three design goals of a dynamic control parameter configuration mechanism for I2VFC.

1. Automatic configuration of I2VFC control parameters
2. Adaptability of I2VFC control parameters to network changes
3. High scalability in terms of the number of PE routers and the number of VPN flows

In this paper, we propose a dynamic control parameter configuration mechanism called *DCPC (Dynamic Control Parameter Configuration)* by introducing the concept of a virtual VPN flow called *nominal VPN flow*. With the concept of the nominal VPN flow, a PE router can autonomously configure I2VFC control parameters. Our proposed DCPC periodically configures I2VFC control parameters so that I2VFC realizes a constant ratio between the throughput of a VPN flow and that of the nominal VPN flow. Our proposed DCPC has high scalability in terms of the number of PE routers and the number of VPN flows accommodated in a PE router because the DCPC algorithm is one of distributed algorithms that each PE router operates autonomously.

Through several simulation experiments, we quantitatively evaluate effectiveness of the proposed dynamic control parameter configuration mechanism DCPC. Specifically, we evaluate how accurately and on what timescale fairness among VPN flows is achieved using our dynamic control parameter configuration mechanism DCPC. We show that I2VFC with DCPC can achieve fairness with high accuracy and that I2VFC control parameters can be configured according to network changes on the timescale of approximately 100 times of VPN flow's round-trip time.

The structure of this paper is as follows. First, Section 2 introduces overview of IP-VPN fairness control mechanism I2VFC. Section 3 explains the design goals and the algorithm of our dynamic control parameter configuration mechanism DCPC proposed in this paper. Section 4 quantitatively evaluates the effectiveness of our proposed dy-

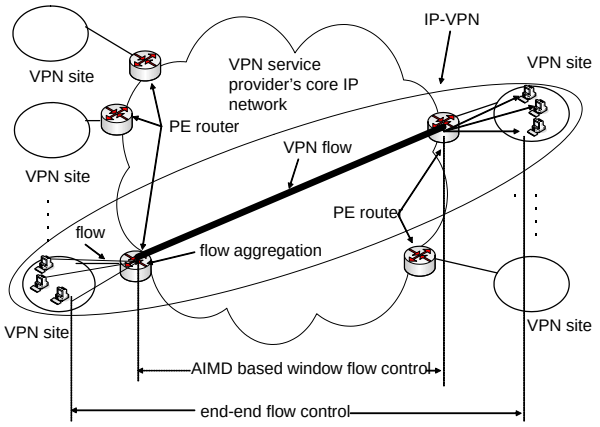

Figure 1: Overview of I2VFC (Inter- and Intra-VPN Fairness Control)

namic control parameter configuration mechanism DCPC through simulation experiments. Finally, Section 5 concludes this paper and discusses future works.

2 I2VFC (Inter- and Intra-VPN Fairness Control)

This section presents overview of IP-VPN fairness control mechanism I2VFC. Refer to [4] for the details of I2VFC.

Figure 1 shows the overview of I2VFC. The core of I2VFC is an AIMD window flow control [6] that operates among IP-VPN service provider's edge routers. Specifically, multiple flows accommodated in the same VPN are aggregated into a single VPN flow and stored in a logical queue for each VPN flow in the ingress PE router. Then, the round-trip time and the packet loss rate of VPN flows are periodically measured by exchanging management packets between ingress and egress PE routers. Based on these information, the ingress PE router performs the AIMD window flow control for adjusting the number of packets injected into the core network.

I2VFC can achieve arbitrary fairness criteria among VPN customers (inter-VPN fairness); that is, the ratio of VPN flow throughputs can be arbitrary controlled by the service provider. The weight of the throughput of VPN flow i $(1 \leq i \leq N)$ is denoted by r_i and the throughput of VPN flow i is by T_i. I2VFC can realize

$$\frac{T_i}{r_i} = \frac{T_j}{r_j} \tag{1}$$

for all i, j $(i \neq j)$.

To achieve inter-VPN fairness, I2VFC needs to appropriately configure parameters of its window flow control (i.e. additive increase factor a and multiplicative decrease factor b) based on the measured round-trip time, the measured packet loss rate and the weight r_i of VPN flow i. The additive increase factor a of VPN flow i is denoted by a_i, and the multiplicative decrease factor b of VPN flow i is by b_i. Furthermore, the round-trip time of VPN flow i is denoted by R_i, and the packet loss rate of VPN flow i is by p_i. Then the ratio η for VPN flow throughput is approximately given by

$$\eta = \frac{T_i}{T_j} \tag{2}$$

$$\simeq \sqrt{\frac{a_i b_j (2 - b_i)}{a_j b_i (2 - b_j) \gamma^2 \delta}} \, . \tag{3}$$

Here, we define $\gamma \equiv R_i / R_j$ and $\delta \equiv p_i / p_j$. It is necessary to configure the additive increase factor a and the multiplicative decrease factor b of each VPN flow so that the value η in Eq. (3) is equal to the ratio r_i / r_j specified by the service provider. If I2VFC control parameters are appropriately configured as $\eta = r_i / r_j$, I2VFC can achieve inter-VPN fairness with high accuracy [4].

Furthermore, I2VFC achieves not only inter-VPN fairness but also fairness among flows accommodated in the same VPN (intra-VPN fairness). Intra-VPN fairness is achieved by simply relying on TCP's congestion control mechanism operating between end hosts. That is, I2VFC itself does not perform any control for achieving intra-VPN fairness. The congestion control mechanism of TCP achieves sufficient intra-VPN fairness because flows accommodated in the same VPN will have the same round-trip time and the same packet loss rate [4]. Since the congestion control mechanism of TCP tries to achieve intra-VPN fairness, it is necessary that the AIMD window flow control operating for each VPN flow does not interfere with the congestion control mechanism of TCP. I2VFC's window flow control avoids such interference by operating at a much larger timescale than the round-trip time of TCP.

3 Dynamic Control Parameter Configuration Mechanism

This section describes the design goals of a dynamic control parameter configuration mechanism for I2VFC and then explains the algorithm of our proposed dynamic control parameter configuration mechanism DCPC (Dynamic Control Parameter Configuration).

3.1 Design goals

1. *Automatic configuration of I2VFC control parameters*

   The first design goal of a dynamic control parameter configuration mechanism is automatically configuring all I2VFC control parameters. I2VFC measures the round-trip time and the packet loss rate at an ingress PE router. Based on these information and the weight r of a VPN flow specified by a service provider in advance, the additive increase factor a and the multiplicative decrease factor b need to be automatically configured so that inter-VPN fairness is achieved. Since IP-VPN is a best-effort network, variation in background traffic and network routing affect the round-trip time and the packet loss rate between ingress and egress PE routers. Hence, it

is desirable that a dynamic control parameter configuration mechanism for I2VFC autonomously detects network changes and appropriately configures the control parameters without necessity of network administrator's intervention.

2. *Adaptability of I2VFC control parameters to network changes*

   The second design goal is that I2VFC control parameter can be configured according to network changes. The timescale at which a dynamic control parameter configuration mechanism for I2VFC configures the I2VFC control parameters is important. Intra-VPN fairness is achieved by relying on TCP's congestion control mechanism between end hosts. This congestion control operates at a timescale of approximately round-trip time. On the contrary, to achieve inter-VPN fairness, the AIMD window flow control of I2VFC is performed between PE routers. The timescale of the AIMD window flow control should be larger than the round-trip time of TCP [4].

   If a dynamic parameter configuration mechanism for I2VFC operates at a timescale smaller than the timescale of the AIMD window flow control of I2VFC and/or the round-trip time of TCP flows, the AIMD window flow control of I2VFC may become unstable. Hence, it is necessary for a dynamic control parameter configuration mechanism to operate at a larger timescale than the timescale of the AIMD window flow control. However, if the timescale at which a dynamic control parameter configuration mechanism for I2VFC configures I2VFC control parameters is too large, the mechanism will not be able to quickly configure them after network changes. Consequently, fairness among VPN customers will be significantly degraded. Therefore, it is thought that not only the inter-VPN fairness in steady state but the inter-VPN fairness in transient state is one of the important performance metrics. The timescale at which a dynamic control parameter configuration mechanism for I2VFC configures control parameters needs to be determined by taking account of the stability and the convergence speed of I2VFC.

3. *High scalability in terms of the number of PE routers and the number of VPN flows*

   The third design goal is that the dynamic parameter configuration mechanism for I2VFC achieves high scalability in terms of the number of PE routers and the number of VPN flows. Currently, customers of IP-VPN services are generally certain organizational units such as companies and groups, so the number of VPNs managed by an IP-VPN service provider is rather small. In the future, however, customers will be an individual user, so that the number of VPNs managed by the IP-VPN service provider will explode. Hence, it is important that a dynamic parameter configuration mechanism for I2VFC also has high scalability in terms of the number of PE routers and the number of VPNs flows.

3.2 Overview

As described in Section 2, in order for I2VFC to achieve inter-VPN fairness, I2VFC control parameters need to be configured so that $\eta = r_i/r_j$ in Eq. (3) is satisfied for any combination of VPN flows. However, as can be seen from (3), it is not trivial to calculate appropriate additive increase factor a and appropriate multiplicative decrease factor b even if weights r of VPN flows specified by a service provider are known.

To achieve the first design goal, it is necessary to estimate the round-trip time R and the packet loss rate p of each VPN flow, because they are required to calculate Eq. (3). In I2VFC, the round-trip time and the packet loss rate for each VPN flow are measured between PE routers by exchanging management packets [4]. However, those measured values cannot be used as-is because those values are measured approximately at the timescale of round-trip time.

Moreover, I2VFC measures only the round-trip time and the packet loss rate of VPN flows among PE routers. I2VFC needs some signaling mechanism between PE routers to measure the round-trip time and the packet loss rate of VPN flows accommodated in other PE routers. For example, there might be a centralized algorithm that configures I2VFC control parameters according to the status of the network using Eq. (3). In such a centralized algorithm, a VPN management server should gather current states of all VPN flows (e.g., the round-trip time and the packet loss rate) from all ingress PE routers. Based on these information, a VPN management server calculates the additive increase factor a and the multiplicative decrease factor b that satisfy $\eta = r_i/r_j$ in Eq. (3), and periodically informs each ingress PE router of the calculated parameters. However, with a centralized algorithm, if there are many PE routers and/or many VPN flows, it is not permissible for the VPN management server to gather states of all VPN flows and inform all ingress PE routers of control parameters. For this reason, it is difficult to satisfy the third design goal.

This paper proposes a dynamic control parameter configuration mechanism DCPC (Dynamic Control Parameter Configuration) for each PE router to configure I2VFC control parameters automatically. Figure 2 shows overview of our proposed dynamic control parameter configuration mechanism DCPC.

DCPC introduces the concept of a virtual VPN flow called *nominal VPN flow* to configure I2VFC control parameters. The nominal VPN flow is a virtual VPN flow that does not exist but all PE routers know all control parameters of the nominal VPN flow. Specifically, a service provider configures parameters (a^*, b^*, R^*, p^*, and r^*) of the nominal VPN flow, which are known by all PE routers.

The basic idea of the dynamic control parameter configuration mechanism DCPC is that a ingress PE router periodically configures I2VFC control parameters considering only the ratio between throughput of the nominal VPN flow and that of the VPN flow accommodated in the PE router. With the concept of the nominal VPN flow, an ingress PE router can configure I2VFC control parameters based on the round-trip time, the packet loss rate, and the weight of VPN flow throughput accommodated in the

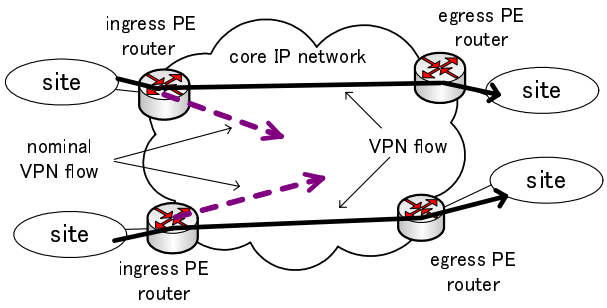

Figure 2: Overview of the dynamic control parameter configuration mechanism DCPC

ingress PE router. This idea enables automatic and distributed operation of each PE router for satisfying the first design goal (automatic configuration) and the third design goal (high scalability).

To achieve the second design goal (adaptability), control parameters of the nominal VPN flow and all VPN flows need to be appropriately configured. It is because the additive increase factor a and the multiplicative decrease factor b of the AIMD window flow control affect transient performance since these factors are roughly correspond to gains of a feedback control system. As shown in Eq. (3), the additive increase factor a and the multiplicative decrease factor b of VPN flows heavily depend on parameters of the nominal VPN flow. Combinations of the additive increase factor a and the multiplicative decrease factor b that satisfy $\eta = r_i/r_j$ in Eq. (3) are infinite, and a choice of parameters determines the transient performance.

Our proposed dynamic control parameter configuration mechanism DCPC, therefore, configures control parameters so that the timescale of the AIMD window flow control of each VPN flow is larger than the timescale of the congestion control of TCP operating between end hosts. The second design goal (adaptability) is achieved by configuring I2VFC control parameters so that the timescale of the AIMD window flow control of each VPN flow is smaller than that of network changes.

3.3 Algorithm

The algorithm of the dynamic parameter configuration mechanism DCPC is a distributed algorithm that realizes automatic operation of PE routers, and consists of two types of operations.

1. Estimate the round-trip time R and the packet loss rate p of each VPN flow

   An ingress PE router estimates the round-trip time and the packet loss rate of each VPN flow accommodated in the ingress PE router. In I2VFC, the ingress PE router measures the round-trip time and the packet loss rate of each VPN flow by exchanging management packets with the egress routers. The ingress PE router calculates average value of the round-trip time and the packet loss rate every χ round-trip time intervals. The n-th measured values of the round-trip time are denoted by R_n and the n-th measured packet loss rate is by p_n. In DCPC, the ingress PE router estimates the round-trip time R and the packet loss rate p by calculating exponential moving average of those measured values:

   $$R \leftarrow w_R R_n + (1 - w_R)R \quad (4)$$
   $$p \leftarrow w_p p_n + (1 - w_p)p \quad (5)$$

   where w_R and w_p are weights of the exponential moving average of the round-trip time and the packet loss rate. As described in Section 3.1, DCPC needs to operate at a larger timescale than the round-trip time of VPN flows. DCPC therefore uses not instantaneous values measured but smoothed values calculated by the exponential moving averages.

2. Periodically configure the additive increase factor a and the multiplicative decrease factor b of each VPN flow

   The ingress PE router configures the additive increase factor a and the multiplicative decrease factor b of each VPN flow accommodated in the router every T intervals. Specifically, the PE router configures the additive increase factor a and the multiplicative decrease factor b based on the estimated round-trip time R and the estimated packet loss rate p of each VPN flow, so that the ratio between throughput of the nominal VPN flow and throughput of a VPN flow satisfies r/r^*. The PE router calculates the additive increase factor a and the multiplicative decrease factor b that satisfy Eq. (3). Namely,

   $$\frac{r}{r^*} = \sqrt{\frac{ab^*(2-b)}{a^*b(2-b^*)\gamma^2\delta}} \quad (6)$$

   where $\gamma \equiv R/R^*$ and $\delta \equiv p/p^*$.

   As described in Section 3.2, the number of combinations of the additive increase factor a and the multiplicative decrease factor b that satisfy Eq. (3) are infinite, and a choice of parameters determines the transient performance. I2VFC control parameters configured by DCPC satisfy $a < 1$ and $b < 0.5$ to avoid interference with the congestion control mechanism of TCP operating between end hosts. Let a_0 and b_0 be the additive increase factor a and the multiplicative decrease factor b at which the timescale of AIMD window flow control is approximately 100 times of the round-trip time. Each PE router chooses the additive increase factor a and the multiplicative decrease factor b that satisfy $a > a_0$ and $b > b_0$. In DCPC, each PE router calculates the average $\overline{a}$ of all a's that satisfied Eq. (6) for $a_0 < a < 1$ and $b_0 < b < 0.5$, and determines $\overline{b}$ by solving Eq. (6) with $\overline{a}$ for b.

4 Simulation

In this section, through simulation experiments, we confirm the proposed dynamic control parameter configuration mechanism DCPC achieves two design goals, i.e.,

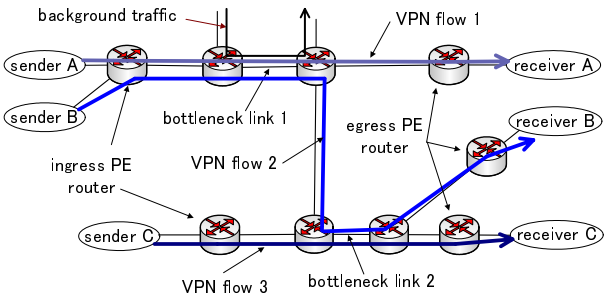

Figure 3: Network topology used in simulation

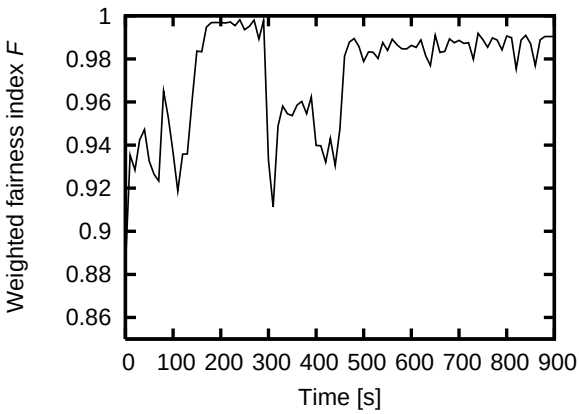

Figure 4: Evolution of the weighted fairness index F

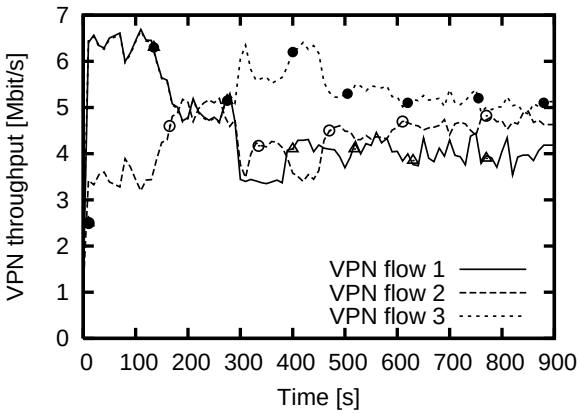

Figure 5: Evolution of VPN throughput

(1) automatic configuration of I2VFC control parameters, (2) adaptability of I2VFC control parameters to network changes. Specifically, we evaluate how accurately and on what timescale inter-VPN fairness is realized by using DCPC.

A weighted fairness index F defined by the following equation [7, 8] is used as a performance index for inter-VPN fairness.

$$F = \frac{(\sum_i^N \frac{T_i}{r_i})^2}{N \sum_i^N (\frac{T_i}{r_i})^2} \quad (7)$$

T_i is the throughput of i-th flow, r_i is the weight of the i-th flow, and N is the number of VPN flows in the network. The weighted fairness index F takes a value between 0 and 1, with $F = 1$ when fairness is completely achieved and with F close to 0 when fairness is not achieved. We calculate the weighted fairness index F every 10 [s] from simulation results. We then evaluate effectiveness of our proposed DCPC by focusing on evolutions of F.

We do not evaluate intra-VPN fairness through simulation experiments since it is known that I2VFC can achieve intra-VPN fairness if the additive increase factor a and the multiplicative decrease factor b satisfy $a < 1$ and $b < 0.5$ [4]. Intra-VPN fairness will be achieved because our proposed dynamic control parameter configuration mechanism DCPC chooses I2VFC control parameters that satisfy $a < 1$ and $b < 0.5$.

Figure 3 shows the network topology used in simulation experiments. In all simulation experiments, the bandwidth of links indicated by *bottleneck link 1* and *bottleneck link 2* are set to 10 [Mbit/s], and the bandwidth of all other links are to 10 [Gbit/s].

Sender hosts continuously send TCP packets to the receiving hosts. VPN flow 1 and VPN flow 2 share the bottleneck link 1, and VPN flow 2 and VPN flow 3 share the bottleneck link 2. The buffer sizes of all routers are 50 [packet]. The weighs of all VPN flows are $r = 3$; i.e., the fairness is achieved when the throughput of all VPN flows are identical.

UDP traffic is generated on the bottleneck link as background traffic. The average arrival rate of background traffic is 20% of the bottleneck link bandwidth, and the packet length is fixed at 1,500 [byte]. The inter-packet time is exponentially distributed. The simulation time is 900 [s], and each simulation is repeated 10 times. In all simulation results, the 95% confidence interval for the weighted fairness index F is within 1% of all of the measurements, so the confidence interval is not shown in the following results. We use OPNET Modeler 9.1A [9] for the simulation experiments.

The lower limits of the additive increase factor and the multiplicative decrease factor, a_0 and b_0, are $a_0 = 0.1$ and $b_0 = 0.01$. The parameters of the nominal VPN flow are as follows: $a^* = 0.5$, $b^* = 0.1$, $R^* = 0.2$ [s], $p^* = 0.008$, and $r^* = 1$. The initial values of a and b are $a = 0.2$ and $b = 0.2$, and other parameters of T, χ, w_R, and w_p are $T = 1000R$, $\chi = 100$, $w_R = 0.6$, and $w_p = 0.6$.

We show simulation results in Figs. 4 through 7. Evolution of the fairness index F for inter-VPN fairness are plotted in Fig. 4, and evolution of the throughput of VPN flows are plotted in Fig. 5. In Fig. 5, symbols (△, ○, ●) represent the time when DCPC updates I2VFC control parameters. In addition, evolution of the round-trip time are plotted in Fig. 6, and evolution of the packet loss rate are plotted in Fig. 7.

Figures 4 and 5 indicate that the fairness index F for inter-VPN fairness is small at the beginning of simulation because I2VFC control parameters are not appropriate. However, inter-VPN fairness is achieved after approximately 40 [s] (approximately 250 times of the round-trip time of VPN flow 2) since control parameters of each VPN

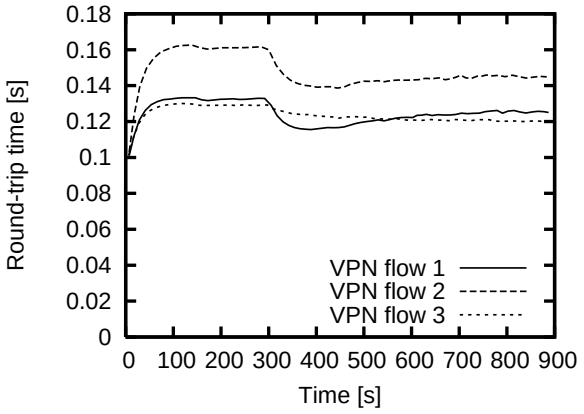

Figure 6: Evolution of the round-trip time

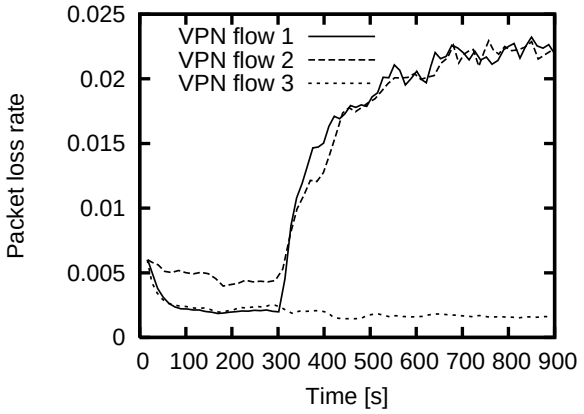

Figure 7: Evolution of the packet loss rate

flow are appropriately configured at $t = 135$ [s]. When the background traffic is initiated at $t = 300$ [s], the round-trip time and the packet loss rate are changed (see Figs. 6 and 7). The fairness index F for inter-VPN fairness is small around $t = 300$[s] because I2VFC control parameters are not appropriate. However, inter-VPN fairness is achieved after approximately 70 [s] (approximately 480 times of the round-trip time of VPN flow 2) since control parameters of each VPN flow are appropriately configured at $t = 389$ [s].

From these observations, we conclude that I2VFC with the proposed DCPC can achieve fairness with extremely high accuracy in various network configurations and that I2VFC control parameters can be configured according to network changes at the timescale of approximately 100 times of the round-trip time.

5 Conclusion

In this paper, we have proposed a dynamic control parameter configuration mechanism DCPC (Dynamic Control Parameter Configuration) that dynamically configures I2VFC control parameters according to network changes. DCPC utilizes the concept of a virtual VPN flow called *nominal VPN flow* for configuring I2VFC control parameters automatically.

Through simulation experiments, we have shown that I2VFC with DCPC can achieve fairness with extremely high accuracy in various network configurations, and I2VFC control parameters can be configured according to network changes at the timescale of approximately 100 times of the round-trip time.

Our future work includes tuning of other control parameters such as a_0, b_0, T, χ, w_R, and w_p, and parameters of the nominal VPN flow for optimizing the performance of DCPC.

References

[1] B. Gleeson *et al.*, "A framework for IP based virtual private networks," *Request for Comments (RFC) 2764*, Feb. 2000.

[2] M. Carugi and D. McDysan, "Service requirements for layer 3 provider-provisioned virtual private networks (PPVPNs)," *Request for Comments (RFC) 4031*, Apr. 2005.

[3] A. Nagarajan, "Generic requirement for provider-provisioned virtual private networks (PPVPN)," *Request for Comments (RFC) 3809*, June 2004.

[4] O. Honda, H. Ohsaki, M. Imase, J. Murayama, and K. Matsuda, "Scalable IP-VPN flow control mechanism supporting arbitrary fairness criteria — part 1: architecture design —," *to be presented at Fourteenth International Conference on Computer Communications and Networks*, Aug. 2005.

[5] R. Callon and M. Suzuki, "A framework for layer 3 provider-provisioned virtual private networks PPVPNs," *Request for Comments (RFC) 4110*, July 2005.

[6] D.-M. Chiu and R. Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks," *Computer Networks and ISDN Systems*, vol. 17, pp. 1–14, June 1989.

[7] R. Pletka, A. Kind, M. Waldvogel, and S. Mannal, "Closed-loop congestion control for mixed responsive and non-responsive traffic," in *Proceedings of IEEE GLOBECOM 2003*, pp. 4180–4186, Dec. 2003.

[8] R. Jain, *The Art of Computer Systems Performance Analysis*. New York: Wiley-Interscience, Apr. 1991.

[9] Opnet Technologies, Inc., "OPNET." `http://www.opnet.com/`.