Fluid-Based Analysis of a Network with DCCP Connections and RED Routers

Hiroyuki Hisamatsu, Hiroyuki Ohsaki, Masayuki Murata
Graduate School of Information Science and Technology, Osaka University
1-5 Yamadaoka, Suita, Osaka 565-0871, Japan
E-mail: {hisamatu,oosaki,murata}@ist.osaka-u.ac.jp

Abstract

In this paper, we model DCCP congestion control mechanism and RED as independent discrete-time systems using fluid-flow approximation. By interconnecting DCCP connections and RED routers, we model the entire network as a feedback system called DCCP/RED. *We then analyze the steady state performance and the transient state performance of DCCP/RED. Specifically, we derive the packet transmission rate of DCCP connections, the packet transmission rate, the packet loss probability, and the average queue length of the RED router in steady state. Moreover, we investigate the parameter region where DCCP/RED operates stably by linearizing DCCP/RED around its equilibrium point. We also evaluate the transient state performance of DCCP/RED in terms of ramp-up time, overshoot, and settling time. Consequently, we show that the stability and the transient state performance of DCCP/RED degrade when the weight of the exponential weighted moving average, which is one of RED control parameters, is small. To solve this problem, by adding changes to the function with which RED determines the packet loss probability, we propose RED-IQI (RED with Immediate Queue Information). We analyze the transient state performance of the feedback system DCCP/RED-IQI where DCCP connections and RED-IQI routers are interconnected. Consequently, we show that DCCP/RED-IQI has significantly better transient state performance than DCCP/RED.*

1. Introduction

In recent years, real-time applications, such as video streaming, IP telephone, TV conference, and network game, become popular rapidly by increasing speed of the network or rising demand for multimedia applications. DCCP (Datagram Congestion Control Protocol) is proposed as a new transport layer protocol for real-time applications [1]. DCCP performs congestion control between source and destination hosts, and an application using DCCP can choose the type of congestion control mechanisms. Currently, *TCP-like congestion control profile*that performs congestion control similar to TCP, and *TFRC congestion control profile* that performs congestion control similar to TFRC (TCP Friendly Rate Control) are proposed. Whereas DCCP performs congestion control between source and destination hosts, AQM (Active Queue Management) mechanisms that perform congestion control at routers in the network have been capturing the spotlight in recent years [2]. A representative AQM mechanism is RED (Random Early Detection) [3], which probabilistically discards an arriving packet.

In the literature, many studies on the congestion control mechanism of TCP, which is adopted in the TCP-like congestion control profile of DCCP, have been extensively performed [4-8]. In particular, characteristics of the mixed environment of TCP connections and RED routers have been extensively studied. For instance, in [7], the congestion control mechanism of TCP and RED are modeled as independent discrete-time systems. The entire network is then modeled as a feedback system where TCP connections and the RED router are interconnected. By applying control theory, the steady state performance and the transient state performance of the TCP congestion control mechanism and RED are analyzed. Moreover, in [4-6], the TCP congestion control mechanism and RED are modeled as independent continuous-time systems, and the steady state performance of RED is analyzed. In [8], it is shown that the transient state performance and the robustness of RED improve, when the function with which RED determines the packet loss probability is changed to a concave function to the average queue length.

Although characteristics of the mixed environment of TCP congestion control mechanism and RED have been sufficiently investigated, characteristics of the mixed environment of TFRC congestion control mechanism and RED have not been sufficiently studied [9-12]. In [11], fairness between TCP-friendly rate control mechanism and TCP in steady state is evaluated with simulations and traffic measurements of the Internet. Moreover, in [10], fairness between TFRC and TCP is evaluated by simulation. The

transient state performance of a TCP-friendly rate control mechanism is also evaluated. However, these studies assume that all routers are DropTail routers and the effect of the interaction between TFRC connections and RED routers has not been fully investigated [9, 12].

In this paper, we model DCCP congestion control mechanism and RED as independent discrete-time systems by using the modeling approach in [7]. We then analyze the steady state performance and the transient state performance of DCCP/RED. Specifically, we derive the packet transmission rate of DCCP connections, the packet transmission rate, the packet loss probability, and the average queue length of the RED router in steady state. Moreover, we investigate the parameter region where DCCP/RED operates stably by linearizing DCCP/RED around its equilibrium point. We also evaluate the transient state performance of DCCP/RED in terms of ramp-up time, overshoot, and settling time.

The organization of this paper is as follows. In Section 2, we model DCCP congestion control mechanism and RED as independent discrete-time systems. By interconnecting these models, we obtain DCCP/RED model, the model of the entire network. In Section 3, we derive the packet transmission rate of DCCP connections, the packet transmission rate, the packet loss probability, and the average queue length of the RED router in steady state. In Section 4, we analyze the transient state performance of DCCP/RED by linearizing DCCP/RED around its equilibrium point. Moreover, in Section 5, we present several numerical examples and show quantitatively how the steady state performance and the transient state performance of DCCP/RED change with the bottleneck link bandwidth and the propagation delay of the network. We also show that the stability and the transient state performance of DCCP/RED degrade when the weight of the exponential weighted moving average is small. In Section 6, we propose RED-IQI by adding changes to the function with which RED determines the packet loss probability. We then analyze the transient state performance of DCCP/RED-IQI. Finally, in Section 7, we conclude this paper and discuss future works.

2. Modeling DCCP and RED

In this paper, we model DCCP congestion control mechanism and RED as independent discrete-time systems with a time slot of Δ. We model the entire network as a single feedback system where DCCP connections and RED routers are interconnected. First, we model the congestion control mechanism of DCCP as a discrete-time system, where the input is the packet arrival rate at a destination host and the output is the packet transmission rate from a source host. Next, we model RED as a discrete-time system, where the input is the packet arrival rate and the output is the packet transmission rate.

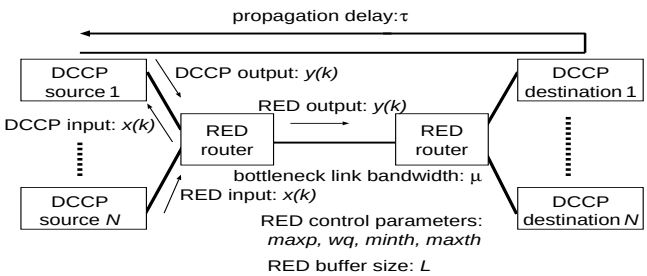


Figure 1. Analytic model

Figure 1 shows the analytic model used in this paper. N DCCP connections share the single bottleneck link. All DCCP connections' two-way propagation delays are equal, which are denoted by τ. The bottleneck link bandwidth is denoted by μ. We denote four control parameters of RED by max_p (maximum packet loss probability), max_{th} (maximum threshold), min_{th} (minimum threshold), and w_q (weight of exponential weighted moving average). Furthermore, RED buffer size is denoted by L.

In this analysis, we introduce a concept of *the packet arrival rate at a destination host* notified of a source host by ACK packets, to unify the input and the output of the models to the packet arrival/transmission rate. Since information on the arrival status of packets at a destination host is included in ACK packets, a source host can estimate the packet arrival rate at a destination host.

In this analysis, we assume the followings; since DCCP is mainly used for real-time applications, it is assumed that a source host always has data to transfer. When the packet loss probability of the network is small and DCCP congestion control works appropriately, DCCP operates in the congestion avoidance phase. Therefore, DCCP with the TCP-like congestion control profile is assumed to operate in the congestion avoidance phase.

First, we model change of the DCCP window size. The packet loss probability in the network is denoted by p, and the DCCP window size is denoted by w. Change of the DCCP window size is given by [13]

$$\begin{aligned} w \quad \leftarrow \quad & w + (1-p)\tfrac{1}{w} - p\,(1 - p_{TO}(w,p))\,\tfrac{1}{2}\tfrac{4\,w}{3} \\ & -p\,p_{TO}(w,p)\left(\tfrac{4\,w(k)}{3} - 1\right), \end{aligned}$$

where $p_{TO}(w,p)$ is the probability that DCCP detects the packet loss by the timeout mechanism when the window size is w and the packet loss probability is p [14]:

$$p_{TO}(w,p) = \frac{(1-(1-p)^3)\,(1+(1-p)^3\,(1-(1-p)^{w-3}))}{(1-(1-p)^w)}.$$

$p(k)$ is defined as the packet loss probability at slot k in the network, $R(k)$ the DCCP round-trip time, and $w(k)$

the DCCP window size. The packet loss probability of the network that a source host detects at slot k is given by $p(k - \frac{R(k)}{\Delta})$. Suppose that ACK packets are not discarded due to congestion on the path from a destination host to a source host, the ACK Ratio value converges to 1Hence, the DCCP window size $w(k+1)$ at slot $k+1$ is approximately given by

$$\begin{aligned} w(k+1) \simeq w(k) + \frac{w(\delta)}{R(k)}\,\Delta\Big\{&(1-p(\delta))\frac{1}{w(k)} \\ &-p(\delta)(1-p_{TO}(w(\delta),p(\delta)))\tfrac{2\,w(k)}{3} \\ &-p(\delta)\,p_{TO}(w(\delta),p(\delta))(\tfrac{4\,w(k)}{3}-1)\Big\}, \end{aligned} \tag{1}$$

where $\delta \equiv k - R(k)/\Delta$.

The packet arrival rate at a destination host $x(k)$ is determined by the past packet transmission rate of a source host and the past packet loss probability in the network, $y(\delta)$ and $p(\delta)$.

$$x(k) \quad = \quad (1-p(\delta))y(\delta)$$

Thus, the DCCP packet transmission rate is given by the following equation from change of the DCCP window size given by Eq. (1).

$$\begin{aligned} &y(k+1) \simeq f(x(k), y(k), R(k)) \\ &= \quad y(k) + \Delta\tfrac{x(k)}{y(k)R(k)^2} - \tfrac{2}{3}\Delta y(k)\,\{y(\delta)-x(k)\} \\ &\quad \times\Big\{1 - p_{TO}(x(\delta)\,R(\delta), 1-\tfrac{x(k)}{y(\delta)})\Big\} \\ &\quad -\Delta\Big\{\tfrac{4}{3}y(k) - \tfrac{1}{R(k)}\Big\}\,\{y(\delta)-x(x)\} \\ &\quad \times p_{TO}(x(\delta)\,R(\delta), 1-\tfrac{x(k)}{y(\delta)}) \end{aligned} \tag{2}$$

Next, we model the congestion control mechanism of DCCP with the TFRC congestion control profile as a discrete-time system. The input $x(k)$ of DCCP with the TFRC congestion control profile is the packet arrival rate at the destination host notified of the source host at slot k. Moreover, the output $y(k)$ is the packet transmission rate from a source host at slot k.

The packet loss event rate at slot k is defined by $p_e(k)$, and the DCCP connection's round-trip time $R(k)$. Suppose that the source host receives an ACK packet at slot k. In this case, the DCCP source host changes the transmission rate $y(k+1)$ at slot $k+1$ as [15]

$$y(k+1) \quad = \quad \min\left(X(p_e(k), R(k)), 2\,x(k)\right), \tag{3}$$

where $X(p_e(k), R(k))$ is given by

$$X(p_e(k), R(k)) = \frac{1}{R(k)\sqrt{\frac{2p_e(k)}{3}} + t_{RTO}\left(3\sqrt{\frac{3p_e(k)}{8}}p_e(k)(1+32p_e(k)^2)\right)}, \tag{4}$$

where t_{RTO} is the TCP retransmission timer, and is can be approximated by $4R(k)$ [15].

Supposing that a RED router discards a packet randomly with the probability p, the packet loss event rate p_e measured by DCCP and the packet loss probability p at a RED router satisfy the following relation:

$$\begin{aligned} \tfrac{1}{p(k)} \quad = \quad & 1 \times \textstyle\sum_{i=1}^{M}\left((1-p_e(k))^{i-1}\,p_e(k)\right) \\ & + \textstyle\sum_{i=M+1}^{\infty}\left(i\,(1-p_e(k))^{i-1}\,p_e(k)\right), \end{aligned} \tag{5}$$

where $M(= R(k)\,y(k))$ is the number of packets that arrive at the RED router during a round-trip time.

Finally, we model the RED router as a discrete-time system. The input $x(k)$ is the packet arrival rate at the RED router at slot k. Moreover, the output $y(k)$ is the packet transmission rate from the RED router at slot k.

We define μ as the bottleneck link bandwidth and $p(k)$ as the probability that the RED router discards packets. Since the packet arrival rate at the RED router is $x(k)$, the packet transmission rate from the RED router is given by $(1-p(k))\,x(k)$. Furthermore, since the maximum packet transmission rate from the RED router is limited by the output link bandwidth, the maximum of $y(k)$ is limited by the bottleneck link bandwidth μ. Hence, the output $y(k)$ of RED is given by [13]

$$y(k) = \min((1-p(k))\,x(k), \mu). \tag{6}$$

The current queue length of RED at slot k is denoted by $q(k)$, and the average queue length is denoted by $\overline{q}(k)$. When the buffer size of the RED router is L, the current queue length $q(k+1)$ at slot $k+1$ is given by [13]

$$\begin{aligned} &q(k+1) \\ &\quad = \quad \min\left[\max\left\{q(k) + (x(k)-\mu)\,\Delta, 0\right\}, L\right]. \end{aligned} \tag{7}$$

Let q be the current queue length of RED, and $\overline{q}$ be the average queue length of RED. RED updates the average queue length $\overline{q}$ for every packet receipt as [3]

$$\overline{q} \leftarrow (1-w_q)\,\overline{q} + w_q\,q. \tag{8}$$

Since the packet arrival rate at slot k is $x(k)$, the average queue length $\overline{q}(k)$ at slot $k+1$ is approximately given by [13]

$$\overline{q}(k+1) \quad \simeq \quad \overline{q}(k) + x(k)\,\Delta\,w_q(q(k)-\overline{q}(k)). \tag{9}$$

RED determines the packet loss probability $p_b(k)$ from its average queue length $\overline{q}(k)$ [3] as

$$p_b(k) \quad = \quad \begin{cases} 0 & \text{if } \overline{q}(k) < min_{th} \\ \dfrac{max_p}{max_{th} - min_{th}}(\overline{q}(k) - min_{th}) & \\ & \text{if } min_{th} \le \overline{q}(k) < max_{th} \\ 1 & \text{if } \overline{q}(k) \ge max_{th}. \end{cases} \tag{10}$$

Finally, the RED router discards arriving packets with the probability $p_a(k)$ determined by

$$p_a(k) = \frac{p_b(k)}{1 - count \times p_b(k)}, \tag{11}$$

where $count$ is the number of packets arrived at the router since the last packet discarded. Since the packet loss probability $p(k)$ in the RED router is the average of $p_a(k)$, it is given by [3]

$$p(k) \quad = \quad \frac{2p_b(k)}{1+p_b(k)}. \tag{12}$$

Note that using the current queue length $q(k)$ of RED, a DCCP connection's round-trip time at slot k is given by

$$R(k) \quad = \quad \frac{q(k)}{\mu} + \tau. \tag{13}$$

3. Steady State Analysis

In what follows, we analyze the steady state performance of DCCP/RED utilizing analytic models constructed in Section 2. Specifically, we derive the packet transmission rate of DCCP connections, the packet transmission rate, the packet loss probability, and the average queue length of RED in steady state. In Section 5, we will validate our approximate analysis by comparing numerical examples with simulation ones.

Since the congestion control mechanism of DCCP with the TCP-like congestion control profile is the AIMD window control, the window size oscillates when the feedback delay is not negligible. Consequently, the packet transmission rate never converges to a fixed value. Note that the output from our DCCP model with the TCP-like congestion control profile represents not an instantaneous value of the oscillating packet transmission rate, but the expected value of the packet transmission rate.

The packet transmission rate of DCCP and RED in steady state ($k \rightarrow \infty$) are denoted by y_D^* and y_R^*, respectively. Let N be the number of DCCP connections. We can numerically obtain y_D^* and y_R^* by solving equations $y(k+1) = y(k) = y_R^*$, $x(k) = \frac{y_R^*}{N}$ (Eq. (2)), $y(k+1) = y(k) = y_R^*$, and $x(k) = N\, y_D^*$ (Eq. (6)). Focusing on the input x_R^* and the output y_R^* of a RED router, we have the following relation

$$y_R^* \quad = \quad (1-p^*)\, x_R^*, \tag{14}$$

where p^* is the packet loss probability at the RED router in steady state. We can obtain p^* by solving Eq. (14) for p^*. Furthermore, from Eqs. (10) and (12), we can easily obtain the average queue length $\overline{q}^*$ of the RED router.

4. Transient State Analysis

We then analyze the transient state performance of DCCP/RED by linearizing the discrete-time model around its equilibrium point.

First, we focus on the feedback system where DCCP connections with the TCP-like congestion control profile and RED routers are interconnected. The states of DCCP and RED are determined by the packet arrival rate $x_D(k)$ at the destination host (notified by a destination host via ACK packets) at slot k, the packet transmission rates $y_D(k) \cdots y_D(k - \frac{R(k)}{\Delta})$ from the source host, the packet arrival/transmission rate of the RED router at slot k, $x_R(k)$ and $y_R(k)$. We introduce a state vector $\mathbf{x}(k)$ that is composed of differences between each state variable at slot k and its equilibrium value:

$$\mathbf{x}(k) \quad \equiv \quad \begin{bmatrix} x_D(k) & - & x_D^* \\ y_D(k) & - & y_D^* \\ & \vdots & \\ y_D(k - \frac{R(k)}{\Delta}) & - & y_D^* \\ x_R(k) & - & x_R^* \\ y_R(k) & - & y_R^* \end{bmatrix} \tag{15}$$

We focus on state transition between slot k and slot $k+1$. Although all discrete models (Eqs. (1), (2), (6)–(12)) in our analysis are nonlinear, they can be written in the following matrix form by linearizing them around their equilibrium values x_D^*, y_D^*, x_R^*, and y_R^*.

$$\mathbf{x}(k+1) \quad = \quad \mathbf{A}\mathbf{x}(k), \tag{16}$$

where $\mathbf{A}$ is the state transition matrix of the state vector from $\mathbf{x}(k)$ to $\mathbf{x}(k+1)$. The eigenvalues of the state transition matrix $\mathbf{A}$ determine the transient state performance (i.e., convergence performance to the equilibrium point) of the discrete-time systems given by Eqs. (1), (2), (6)–(12). Let $\lambda_i (1 \leq i \leq \frac{R(k)}{\Delta} + 3)$ be the eigenvalues of the state transition matrix $\mathbf{A}$. The maximum absolute value of eigenvalues (*maximum modulus*) determines the stability and the transient state performance of the feedback system around its equilibrium point [16]. It is known that the smaller the maximum modulus is, the better the transient state performance becomes. It is also known that the system is stable if the maximum modulus is less than 1.0.

Next, we focus on the feedback system where DCCP connections with the TFRC congestion control profile and RED routers are interconnected. The states of DCCP with the TFRC congestion control profile and RED are determined by the packet arrival rate $x_D(k)$ at the destination host at slot k , the packet transmission rates $y_D(k) \cdots y_D(k - \frac{R(k)}{\Delta})$ from the source host, and the packet arrival/transmission rate of RED, at slot k, $x_R(k)$ and $y_R(k)$. Hence, the state vector $\mathbf{x}(k)$ that is composed of differences between each state variable at slot k and its equilibrium value is given by Eq. (17).

We assume that the DCCP destination host sends an ACK packet to its source host every n slot. We focus on state transition between slot k and slot $k+n$. Although all discrete models in our analysis (Eqs. (3)–(12)) are nonlinear, they can be written in the following matrix form by linearizing them around their equilibrium values x_D^*, y_D^*, x_R^*,

and y_R^*.

$$\mathbf{x}(k+n) \quad = \quad \mathbf{A}\,\mathbf{B}^{n-1}\mathbf{x}(k), \tag{17}$$

where $\mathbf{A}$ is the state transition matrix of the state vector from $\mathbf{x}(k)$ to $\mathbf{x}(k+1)$ when the DCCP source host receives an ACK packet (Eq. (3)). Moreover, $\mathbf{B}$ is the state transition matrix of the state vector from $\mathbf{x}(k)$ to $\mathbf{x}(k+1)$ when the DCCP source host does not receive any ACK packet (i.e., $x(k+1) = x(k)$). $\mathbf{A}\,\mathbf{B}^{n-1}$ is the state transition matrix of the state vector from $\mathbf{x}(k)$ to $\mathbf{x}(k+n)$. The eigenvalues of the state transition matrix determine the transient state performance (i.e., the convergence performance to the equilibrium point) of the discrete-time system given by Eqs. (3)–(12).

5. Numerical Examples

In this section, by presenting some numerical examples, we show quantitatively how the steady state performance and the transient state performance of DCCP/RED change according to the bottleneck link bandwidth and the propagation delay of the network. Furthermore, we validate our approximate analysis by comparing analytic results with simulation ones.

We performed simulation using ns-2 for the network topology shown in Fig. 1. In this network, the link between two RED routers is the bottleneck, so that we focus on the packet loss probability and the average queue length of the upstream RED router. We run simulation for 150 [s] and used simulation result of the last 100 [s] for measuring DCCP connections' packet transmission rates and the packet loss probability of the RED router. We repeated simulation 10 times and measured averages of the DCCP connections' packet transmission rates and the packet loss probability of the RED router.

Unless explicitly stated, in the following numerical examples and simulations, we use the following parameters: the number of DCCP connections N is 10, DCCP connection's two-way propagation delay τ is 50 or 100 [ms], the bandwidth of all access links is $10 \times \mu$ [Mbit/s], the buffer size of RED routers L is 250 [packet], and RED control parameters are (max_p, min_{th}, max_{th}, w_q) = (0.1, 20 [packet], 100 [packet], 0.002).

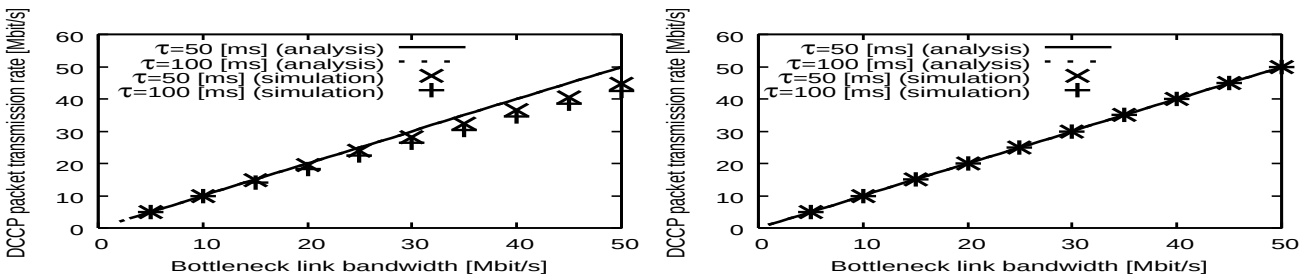
First, we focus on the steady state performance of DCCP/RED. We show the DCCP packet transmission rate for different settings of the bottleneck link bandwidth in Fig. 2. Here, we configure the DCCP connection's two-way propagation delay to $\tau = 50$ and $\tau = 100$ [ms]. Figure 2(a) shows results for DCCP with the TCP-like congestion control profile. Figure 2(b) shows for DCCP with the TFRC congestion control profile. These figures indicate that the DCCP packet transmission rate increases as the bottleneck link bandwidth increases. Moreover, we compare analytic results with simulation ones. In DCCP with the TCP-like congestion control profile, some errors are observed between analytic results and simulation ones in the region where bottleneck link bandwidth is large. In other region, analytic results and simulation ones coincide closely.

The disagreement between analytic and simulation results in DCCP with the TCP-like congestion control profile is probably caused by sensitivity in the control parameter setting of RED. Namely, the control parameter setting of RED becomes inappropriate when the bottleneck link bandwidth is large, so that utilization of the RED router in simulation is degraded.

Next, we focus on the transient state performance of DCCP/RED. Figure 3 shows the maximum modulus of the state transition matrix ($\mathbf{A}$ or $\mathbf{A}\,\mathbf{B}^{n-1}$) of DCCP/RED for different settings of the bottleneck link bandwidth. Figure 3(a) shows results for DCCP with the TCP-like congestion control profile (Eq. (2)). Figure 3(b) shows results for DCCP with the TFRC congestion control profile (Eq. (3)). In these figures, for investigating the effect of a RED control parameter on its transient performance, the weight w_q of the exponential weighted moving average of RED is changed to 0.0002, 0.002 and 0.02. Moreover, the number of DCCP connections is $N = 1$, the two-way propagation delay of DCCP connection is $\tau = 10$ [ms].

These figures show that the maximum modulus increases as the bottleneck link bandwidth increases. This means that the transient state performance of DCCP/RED degrades as the bottleneck link bandwidth increases. Moreover, it can be found that the maximum modulus increases as the weight w_q of the exponential weighted moving average of RED becomes small. This can be explained as follows. The time for the average queue length of RED following change of the network state increases as the weight w_q of the exponential weighted moving average becomes small. Hence, it becomes slow that the packet loss probability of RED follows change of the network state. Namely, setting w_q to be a small value has the same effect with increasing the feedback delay of the entire network.

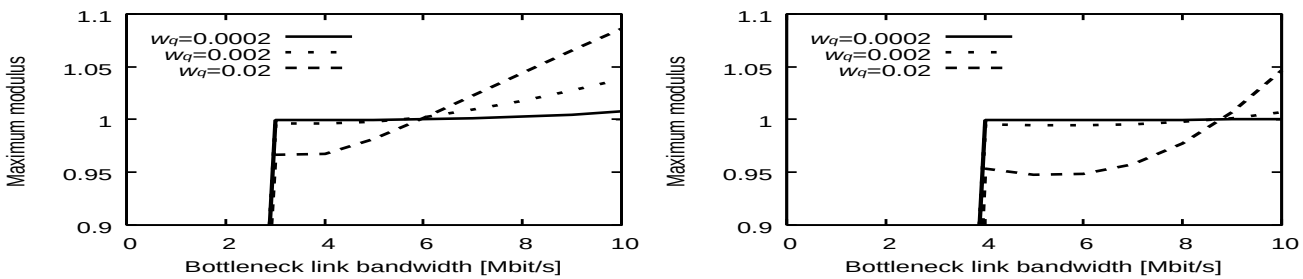
Finally, we investigate how the maximum modulus of the state transition matrix of DCCP/RED affects the transient state performance of DCCP/RED. Table 1 shows the average queue length $\overline{q}^*$, the maximum modulus λ of the state transition matrix of DCCP/RED, and three transient performance metrics: ramp-up time, overshoot and settling time [16]. In our experiments, ramp-up time is defined as the time required for the average queue length of RED to reach 95% of the equilibrium value. Overshoot is defined as the maximum difference of the average queue length of RED from the equilibrium value. Settling time is defined as the time required for the average queue length of RED to be settled within 5% of the equilibrium value. Initial values of state variables are set to 50% of their equilibrium values.



(a) DCCP with TCP-like congestion control profile

(b) DCCP with TFRC congestion control profile

Figure 2. DCCP/RED steady state performance (DCCP packet transmission rate)



(a) DCCP with TCP-like congestion control profile

(b) DCCP with TFRC congestion control profile

Figure 3. DCCP/RED transient state performance (maximum modulus of the state transition matrix)

The weight w_q of the exponential weighted moving average of RED is configured to 0.0002, 0.002 and 0.02. Moreover, the number of DCCP connections is $N = 1$, the bottleneck link bandwidth is $\mu = 4$ [Mbit/s], and the two-way propagation delay of DCCP is $\tau = 10$ [ms].

These results show that ramp-up time, overshoot and settling time decrease small as the weight w_q of the exponential weighted moving average of RED increases. Namely, one can find that the larger the weight w_q of the exponential weighted moving average of RED is, the better the transient performance becomes.

6. RED-IQI (RED with Immediate Queue Information)

In Section 5, we have shown that the transient state performance is degraded as the weight w_q of the exponential weighted moving average becomes small in a system where DCCP connections and RED routers are interconnected.

The packet loss probability p_b of the RED router is determined by the liner function of $(\overline{q} - min_{th})/(max_{th} - min_{th})$ (Eq.10). We call $(\overline{q} - min_{th})/(max_{th} - min_{th})$ *queue occupancy*. Use of this function is determined without sufficiently taking account of the steady state performance and the transient state performance of RED. It is known that when the concave function is used as the function that determines the packet loss probability p_b of the RED router, the transient state performance and the robustness of RED improve [8].

Therefore, in this section, to improve the stability and transient state performance of the system where DCCP connections and RED routers are interconnected, we propose a RED-IQI (RED with Immediate Queue Information) by adding the following changes to RED.

First, we change the calculation method of the average queue length of RED. In RED-IQI, the weight of the exponential weighted moving average is set to $w_q = 1$. Thereby, the feedback delay of DCCP/RED-IQI becomes small, and the stability and the transient state performance are expected to improve. However, by configuring to $w_q = 1$, the packet loss probability of RED-IQI may sensitively fluctuate according to temporary variation of the network state.

w_q	profile	$\overline{q}^*$	λ	ramp-up time [ms]	overshoot [packet]	settling time [ms]
0.0002	TCP-like	51.443	0.9996	920	27.462	27340
0.002	TCP-like	51.443	0.9967	350	23.182	7530
0.02	TCP-like	51.443	0.9678	170	14.330	840
0.0002	TFRC	71.724	0.9995	1520	35.079	31970
0.002	TFRC	71.724	0.9954	530	21.649	4990
0.02	TFRC	71.724	0.9533	250	2.849	250

Table 1. DCCP/RED transient state performance indexes

However, since the AIMD congestion control is used in the TCP-like congestion control profile, it is thought that the variation of the packet loss probability causes little performance degradation. On the other hand, since the TFRC congestion control profile smooths the packet loss event rate [15], it is thought that the variation of the packet loss probability is also causes little performance degradation.

Next, we change the function that determines the packet loss probability of RED. RED determines the packet loss probability using the linear function to the queue occupancy. In RED-IQI, we change this function to a concave function. Specifically, we change the function that determines the packet loss probability p_b to

$$p_b = max_p \, \mathcal{G}_\phi \left(\frac{\overline{q} - min_{th}}{max_{th} - min_{th}} \right), \qquad (18)$$

where $\mathcal{G}_\phi(x)$ is defined as

$$\mathcal{G}_\phi(x) = \left(1 - \sqrt{1 - x^2}\right)^\phi. \qquad (19)$$

$\phi(> 0)$ is a parameter determining the concavity. In order for $\mathcal{G}_\phi$ to be concave,

$$\phi \geq \lim_{x \to 0} \frac{-1 + \sqrt{1 - x^2} + x^2 \sqrt{1 - x^2}}{x^2 \sqrt{1 - x^2}} = \frac{1}{2}. \qquad (20)$$

Next, we show quantitatively how the transient state performance of DCCP/RED-IQI changes with the bandwidth and the propagation delay of the network by presenting several numerical examples. Due to space limitation, we just focus on the transient state performance of DCCP/RED-IQI. Figure 4 shows the maximum modulus of the state transition matrix ($\mathbf{A}$ or $\mathbf{A}\,\mathbf{B}^{n-1}$) of DCCP/RED-IQI for different settings of the bottleneck link bandwidth. Figure 4(a) shows results for DCCP with the TCP-like congestion control profile (Eq. (2)). Figure 4(b) shows results for DCCP with the TFRC congestion control profile (Eq. (3)). For comparison purposes, the maximum modulus of the state transition matrix of DCCP/RED is also shown in the figure. Here, the weight w_q of the exponential weighted moving average of RED is configured to 0.002. Moreover, the number of DCCP connections is $N = 1$, and the two-way propagation delay of DCCP connection is $\tau = 10$ [ms].

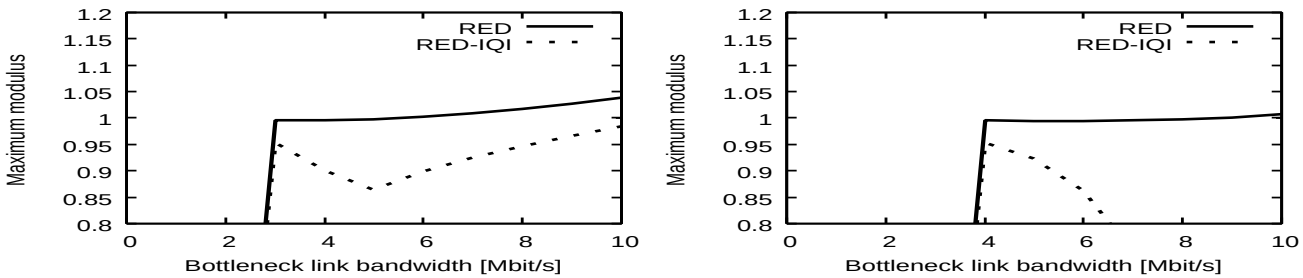
It can be found that the maximum modulus of DCCP/RED-IQI increases as the bottleneck link bandwidth increases from this figure. Moreover, by comparing the maximum modulus of DCCP/RED-IQI with that of DCCP/RED, it can be found that the value of DCCP/RED-IQI is smaller than that of DCCP/RED. This means that DCCP/RED-IQI operates more stably than DCCP/RED.

Table 2 shows the average queue length $\overline{q}^*$, the maximum modulus λ of the state transition matrix, ramp-up time, overshoot and settling time of DCCP/RED-IQI. The number of DCCP connections is $N = 1$, the bottleneck link bandwidth is $\mu = 4$ [Mbit/s], and the two-way propagation delay of DCCP is $\tau = 10$ [ms]. Table 2 shows that the ramp-up time, the overshoot, and the settling time of DCCP/RED-IQI are smaller than those of DCCP/RED (see Tab. 1).

7. Conclusion and Future Work

In this paper, we have modeled DCCP congestion control mechanism and RED as independent discrete-time systems, and have modeled the entire network as a feedback system by interconnecting DCCP connections and RED routers. We have analyzed the steady state and transient state performance of DCCP/RED. We have derived the packet transmission rate of DCCP connections, the packet transmission rate, the packet loss probability, and the average queue length of the RED router in steady state. We have also derived the parameter region where DCCP/RED operates stably by linearizing DCCP/RED model around its equilibrium point. Furthermore, we have evaluated the transient state performance of DCCP/RED in terms of ramp-up time, overshoot, and settling time. Consequently, we have shown that the stability and the transient state performance of DCCP/RED degrade when the weight of the exponential weighted moving average is small. By adding changes to the function with which RED determines the packet loss probability, we propose RED-IQI. We have shown that RED-IQI significantly improves the transient state performance such as overshoot, ramp-up time, and settling time compared with RED.

As future work, it would be interesting to analyze large-



(a) DCCP with TCP-like congestion control profile

(b) DCCP with TFRC congestion control profile

Figure 4. DCCP/RED transient state performance (maximum modulus of the state transition matrix)

	profile	$\overline{q}^*$	λ	ramp-up time [ms]	overshoot [packet]	settling time [ms]
RED-IQI	TCP-like	62.715	0.9011	190	0	190
RED-IQI	TFRC	85.057	0.9525	410	0	410

Table 2. DCCP/RED and DCCP/RED-IQI transient state performance indexes

scale networks by applying the analytic approach proposed in [13] to the DCCP/RED model derived in this paper. It would be also interesting to analyze a network with DCCP connections with different congestion control profiles.

References

[1] E. Kohler, M. Handley, and S. Floyd, "Datagram congestion control protocol (DCCP)," *Internet Draft* `<draft-ietf-dccp-spec-11.txt>`, Mar. 2005.

[2] S. Floyd and K. Fall, "Promoting the use of end-to-end congestion control in the Internet," *IEEE Transactions on Networking*, vol. 7, no. 4, pp. 458–472, May 1999.

[3] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397–413, Aug. 1993.

[4] V. Firoiu and M. Borden, "A study of active queue management for congestion control," in *Proceedings of IEEE INFOCOM 2000*, Mar. 2000, pp. 1435–1444.

[5] C. Hollot, V. Misra, D. Towsley, and W.-B. Gong, "A control theoretic analysis of RED," CMPSCI, Tech. Rep. TR 00-41, July 2000.

[6] ——, "On designing improved controllers for AQM routers supporting TCP flows," in *Proceedings of IEEE INFOCOM 2001*, 2001, pp. 1726–1734.

[7] M. Kisimoto, H. Ohsaki, and M. Murata, "On transient behavior analysis of random early detection gateway using a control theoretic approach," in *Proceedings of the IEEE Control Systems Society Conference on Control Applications (CCA/CACSD 2002)*, Sept. 2002, pp. 1144–1149.

[8] H. Ohsaki and M. Murata, "On packet marking function of active queue management mechanism: Should it be linear, concave, or convex?" in *Proceedings of SPIE's International Symposium on the Convergence of Information Technologies and Communications (ITCom 2004)*, Oct. 2004.

[9] D. Bansal, H. Balakrishnan, S. Floyd, and S. Shenker, "Dynamic behavior of slowly-responsive congestion control algorithms," in *Proceedings of ACM SIGCOMM*, Aug. 2001, pp. 263–274.

[10] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "Equation-based congestion control for unicast applications: the extended version," International Computer Science Institute, Tech. Rep., Mar. 2000.

[11] J. Padhye, J. Kurose, D. Towsley, and R. Koodli, "A model based TCP-friendly rate control protocol," in *Proceedings of NOSSDAV'99*, 1999.

[12] Y. R. Yang, M. S. Kim, and S. S. Lam, "Transient behaviors of TCP-friendly congestion control protocols," *The International Journal of Computer and Telecommunications Networking*, vol. 41, pp. 193–210, Feb. 2003.

[13] H. Ohsaki, J. Ujiie, and M. Imase, "On scalable modeling of TCP congestion control mechanism for large-scale IP networks," in *Proceedings of the 2005 International Symposium on Applications and the Internet (SAINT 2005)*, Feb. 2005, pp. 361–369.

[14] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: a simple model and its empirical validation," in *Proceedings of ACM SIGCOMM '98*, Sept. 1998, pp. 303–314.

[15] M. Handly, S. Floyd, J. Padhye, and J. Widmer, "TCP friendly rate control (TFRC): protocol specification," *Request for Comments (RFC) 3448*, Jan. 2003.

[16] N. S. Nise, *Control Systems Engineering*, 4th ed. New York: John Wiley & Sons, Aug. 2003.