Available online at www.sciencedirect.com
ScienceDirect
Computer Communications 30 (2007) 3778–3786
computer communications
www.elsevier.com/locate/comcom

Dynamics analysis of TCP Veno with RED

Ke Zhang *, Cheng Peng Fu

School of Computer Engineering, Nanyang Technological University, Singapore, Republic of Singapore

Received 29 September 2006; received in revised form 5 September 2007; accepted 5 September 2007
Available online 14 September 2007

Abstract

This paper studies the dynamics of TCP Veno with the queue management of RED (Random Early Detection). We develop a fluid-flow model to describe the behaviors of TCP Veno in wired/wireless networks. This model is further linearized to study TCP Veno's stability issue through the linear feedback control theory. The analysis points out how the RED queue oscillates under different network parameters such as link capacity, round-trip time. Simulations are carried to validate our theoretical analysis. Furthermore, based on the analysis results obtained in this paper, we are able to provide guidelines for tuning RED parameters to stabilize the router queue, and improve the co-existence between TCP Veno and TFRC (TCP-friendly Rate Control) flows.

Keywords: TCP Veno; TCP Reno; RED; TFRC

1. Introduction

TCP is a connection-oriented, reliable and in-order transport protocol. It is known that the current legacy TCP, namely TCP Reno [1], suffers from performance degradation in the wireless networks due to the lack of differentiation between the random and the congestion losses. TCP Veno [2], a sender-side TCP enhancement, was proposed to solve this problem and adopted by Linux Kernel since version 2.6.18 [3]. Veno has the ability to identify network states and adjust the additive increase multiplicative decrease (AIMD) strategy to tackle random losses. Specifically, Veno estimates the number of backlogged packets, N, at a router by

$$N = \left(\frac{cwnd}{BaseRTT} - \frac{cwnd}{RTT}\right) \times BaseRTT$$

where *cwnd* is the TCP congestion window. *BaseRTT* is the minimum of measured round-trip time and reset when packet loss is detected. *RTT* is the actual round-trip time of a tagged packet. Veno compares N with a threshold parameter, β to identify network states. If $N \geqslant \beta$, the network is said to have evolved into a congestive state, and packet loss occurring in this state is considered as a congestion loss. Otherwise, the network is in a non-congestive state, and packet loss occurring in this state is considered as a random loss. In the current Veno implementation, the parameter β is set to 3. Veno adjusts its AIMD algorithm based on the network states,

Multiplicative decrease algorithm:

If $(N < \beta)$ $cwnd = cwnd \cdot 4/5$
else $cwnd = cwnd/2$
Additive increase algorithm:

If $(N < \beta)$ $cwnd += 1/cwnd$ for every new ACK received
else $cwnd += 1/cwnd$ for every other new ACK received

For the random loss, TCP Veno reduces the *cwnd* by a smaller amount (4/5). Ref. [2] shows that, any factor greater than 1/2 but smaller than 1 can be used, so that the cutback in window size is less drastic than the case

* Corresponding author. Tel.: +65 9132 1396.
E-mail addresses: Y030069@ntu.edu.sg (K. Zhang), ascpfu@ntu.edu.sg (C.P. Fu).

0140-3664/$ - see front matter
doi:10.1016/j.comcom.2007.09.004

when the loss is due to congestion. Experiment results prove that the factor of 4/5 is a good setting. More mechanism and experiment details can be seen in [2].

In this paper, we study the dynamics of TCP Veno with the queue management of RED (Random Early Detection) [5]. We make use of the modeling techniques in papers [9,10,12,13] to develop a fluid-flow model of TCP Veno with RED in wired/wireless networks. This model is further linearized around the equilibrium point to study TCP Veno's stability issue. By applying the classical linear feedback control theory, the analytical results reveal how the RED queue oscillation is affected by different network parameters such as link capacity, round-trip time. Simulations are carried to validate our theoretical analysis. Furthermore, based on the analysis results obtained in this paper, we are able to provide guidelines for tuning RED parameters to stabilize the router queue, and improve the co-existence between TCP Veno and TFRC (TCP-friendly Rate Control) flows.

The rest of this paper is organized as follows. In Section 2, we review the fluid-flow model in papers [9,10,12,13] firstly, and then derive the model of TCP Veno. In Section 3, we convert the fluid-flow model to a linear feedback system and apply the linear feedback control theory to analyze its stability issue. In Section 4, simulations are carried to validate our analysis and show how to tune the RED parameters for improving the co-existence of TCP Veno with TFRC. Section 5 concludes this paper and discusses future work.

2. Fluid-flow model of TCP Veno with RED

In this section, we develop the fluid-flow model of TCP Veno with RED in heterogeneous networks. Our model is composed of two parts. The first part: network and RED queue management model is the same as [12,13]. We start with a brief description of them, and then in the second part we model the behavior of TCP Veno's window evolution.

2.1. Network and RED queue management model

The network comprises of L links with capacity $c_l, l \in L$. There are I sources indexed by i. Each source i uses a set of links $Li \subseteq L$. So we have a $L \times I$ routing matrix:

$$M_{li} = \begin{cases} 1 & \text{if } l \in L_i \\ 0 & \text{otherwise} \end{cases}$$

Denote the round-trip time of source i at time t is $R_i(t)$:

$$R_i(t) = T_i + \sum_l M_{li} \frac{q_l(t)}{c_l}$$

where T_i is the round-trip propagation delay of source i and $q_l(t)$ is the instantaneous queue in link l at time t. Denote the loss probability of link l at time t by $p_l(t)$. Assume that $p_l(t)$ is small, then the end-to-end loss probability at time t, $v_i(t)$, is approximated as

$$v_i(t) = \sum_l M_{li} p_l\left(t - R^b_{li}(t)\right)$$

where $R^b_{li}(t)$ is the backward delay from link l to source i. Denote the congestion window size of source i at time t by $w_i(t)$, so the sending rate of source i at time t, $x_i(t)$, is:

$$x_i(t) = \frac{w_i(t)}{R_i(t)}$$

Consider that the last-hop link is wireless link with random loss rate γ_i. Therefore, the aggregate arriving rate of link l at time t is:

$$\begin{aligned} y_l(t) &= \sum_i M_{li} x_i(t - R^f_{li}(t))(1-\gamma_i) \\ &= \sum_i M_{li} \frac{w_i(t - R^f_{li}(t))(1-\gamma_i)}{R_i(t - R^f_{li}(t))} \end{aligned}$$

where $R^f_{li}(t)$ is the forward delay from source i to link l. Note that,

$$R_i(t) = R^f_{li}(t) + R^b_{li}(t)$$

For all links $l \in L$, given the aggregate arriving rate and link capacity, we can calculate the instantaneous queue length $q_l(t)$ derivative ($q_l(t) > 0$) by:

$$\begin{aligned} \dot{q}_l(t) &= y_l(t) - c_l \\ &= \sum_i M_{li} \frac{w_i(t - R^f_{li})(1-\gamma_i)}{R_i(t - R^f_{li})} - c_l \end{aligned} \quad (1)$$

The RED queue management calculates the average queue length $r_l(t)$ by the exponential moving average of instantaneous queue length $q_l(t)$ as follows,

$$r_l(k+1) = (1-\lambda_l) r_l(k) + \lambda_l q_l(k)$$

where λ_l is the queue averaging weight of link l: $0 < \lambda_l < 1$. Consider that the estimate of the average queue length is based on samples taken every $1/c_l$ second, we have,

$$r_l((k+1)/c_l) = (1-\lambda_l) r_l(k/c_l) + \lambda_l q_l(k/c_l)$$

It is useful to convert this equation into a differential equation. Its natural candidate is:

$$\dot{r}_l(t) = A r_l(t) + B q_l(t)$$

Then, in a sampled data system, $r_l(t_{k+1})$ is given by:

$$r_l(t_{k+1}) = \mathrm{e}^{A(t_{k+1}-t_k)} r_l(t_k) + \int_{t_k}^{t_{k+1}} \mathrm{e}^{A(t_{k+1}-\tau)} B d\tau q_l(t_k)$$

Comparing the coefficients, we obtain:

$$1 - \lambda_l = \mathrm{e}^{A/c_l}$$

or

$$A = c_l \log(1-\lambda_l) = -B$$

So, the differential equation can be expressed as,

$$\dot{r}_l(t) = c_l \log(1-\lambda_l) r_l(t) - c_l \log(1-\lambda_l) q_l(t)$$