STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LVI**, Number 3, 2011

A STUDY OF TCP-FRIENDLINESS ON THE SHORT TERM WITH AN APPLICATION TO MEDIA-FRIENDLY CONGESTION CONTROL

ADRIAN STERCA AND ALEXANDRU VANCEA

Abstract. TCP-friendliness is a desired quality of any congestion control algorithm used in the Internet because it expresses fairness towards TCP flows. However, all studies refer to the TCP-friendliness as a long term characteristic of a flow, i.e. they consider only the long-term fairness to TCP. In this paper we introduce the notion of *TCP-friendliness on the short term* and apply it to a multiplicative-type media-friendly congestion control algorithm.

1. Introduction

TCP's AIMD (Additive Increase Multiplicative Decrease) congestion control is not well suited for multimedia streams due to its highly fluctuating throughput. Consequently, other congestion control algorithms which offer a smoother throughput were developed, the so-called TCP-friendly congestion controls algorithms [1]. All these smooth congestion control algorithms have a more stable throughput than TCP's AIMD because they are less aggressive than TCP in using new available bandwidth, but they are also slower responsive to congestion than TCP. Because they offer a more stable throughput, multimedia streams, especially CBR (*Constant Bit Rate*) ones, but also VBR (*Variable Bit Rate*) ones, can be better adapted to predictable bandwidths by the streaming servers. However, although smooth congestion controls improve the delivery of multimedia streams, they are not the optimal solution, because they don't take into consideration media characteristics of the stream (i.e. they are not media-friendly). This led to the development of media-friendly

Received by the editors: May 10, 2011.

2010 *Mathematics Subject Classification.* 68M20, 68U99.

1998 *CR Categories and Descriptors.* C.2.2 [**Computer-Communication Networks**]: Network Protocols – *Protocol architecture (OSI model)*; C.2.3 [**Computer-Communication Networks**]: Network Operations – *Network monitoring*.

Key words and phrases. TCP-friendly congestion control, congestion control, media-friendly congestion control, multimedia streaming.

congestion control [7, 8]. It is important to note that a congestion control algorithm for multimedia streaming must have both characteristics: it must be TCP-friendly (i.e. fair with the network) and it also must be media-friendly to maximize the application benefit.

The contributions of this paper are the following:

- we try to give a classification of media-friendly and TCP-friendly congestion control algorithms
- and we refine the concept of "TCP-friendliness" and distinguish between two types of TCP-fairness: long-term TCP-fairness and short-term TCP fairness.

The rest of the paper is organized as follows. In Section 2 we review TCP-friendly congestion control, focusing on the TCP-Friendly Rate Control (TFRC). Then in Section 3 we define TCP-friendliness on the short term and its utility to media-friendly congestion control algorithms. The paper continues with Section 4 which presents simulations for assessing TCP-friendliness on the short term of UTFRC (Utility-driven TCP-Friendly Rate Control), a multiplicative-type of media-friendly and TCP-friendly congestion control. Finally, the conclusions from Section 5 end the paper.

2. TCP-friendly congestion control

A TCP-friendly congestion control algorithm is a congestion control algorithm which exhibits the same throughput as employed by the TCP's AIMD congestion control algorithm in the same network conditions [1]. This makes TCP-friendly flows fair to TCP flows when consuming network bandwidth. And because TCP-friendly congestion control algorithms typically have a smoother throughput than TCP [2], they are favored over TCP for multimedia streaming applications. There are several proposals for TCP-friendly congestion control [3, 4, 5], but probably the most well known is TCP-Friendly Rate Control (TFRC) [2].

The TCP-Friendly Rate Control [2] is a rate-based congestion control that has two main components: the throughput function and the WALI (i.e., Weighted Average Loss Intervals) mechanism for computing the loss rate. The throughput function is the throughput equation of a TCP-Reno source [6]:

$$X_{TFRC}(p) = \frac{s}{R\sqrt{\frac{2p}{3}} + t_{RTO}(3\sqrt{\frac{3p}{8}})p(1+32p^2)}, \quad (1)$$

where X is the sending rate in bytes/sec, s is the packet size, R is the round-trip time (RTT), p is the steady-state loss event rate and $t_{RTO} = 4 * R$ is the TCP retransmit timeout value. This throughput function is responsible for the TCP-friendliness of TFRC.

The way parameter p from the equation is computed is what gives TFRC its smoothness of throughput. p is computed using WALI as an average loss rate over a time interval including the most recent 8 loss events. The number of packets sent by the sender between two loss events is called a loss interval and the loss rate p is computed using the WALI mechanism as a weighted average of the loss rates in the last 8 loss intervals, where more recent loss intervals get a higher weight [2]:

$$s = \frac{1 * s_0 + 1 * s_1 + 1 * s_2 + 1 * s_3 + 0.8 * s_4 + 0.6 * s_5 + 0.4 * s_6 + 0.2 * s_7}{1 + 1 + 1 + 1 + 0.8 + 0.6 + 0.4 + 0.2}$$

$$p = \frac{1}{s}$$

In the above equation s_i is the length (in packets) of the i-th most recent loss interval, $i \in 0..7$ and the weights are 1, 1, 1, 1, 0.8, 0.6, 0.4, 0.2 starting from the most recent loss interval to the oldest.

Because the loss rate, p, used in the TFRC throughput formula is averaged over time intervals much greater that one round-trip time using the WALI mechanism, the throughput achieved by TFRC is much smoother than the throughput of TCP. Several studies give evidence of this fact [9, 10, 11]. A direct consequence of this is that TFRC reacts slowly (more slowly than TCP) to an increase of the level of congestion in the network, but it reacts also slowly when additional bandwidth becomes available in the network.

Having a throughput smoother than TCP's makes TFRC valuable for multimedia streaming applications because of the increased predictability of its throughput. However, several studies documented some of its limitations [7, 8, 12, 13], its lack of media-friendliness to be more specific.

3. Media-friendly congestion control and TCP-friendliness on the short term

Media-friendly congestion control is a type of congestion control which incorporates also media characteristics like bitrate, buffer fill level, quality measurements etc. in the throughput computing formula besides just network parameters (like round-trip time and loss rate) [7, 8].

To the best of our knowledge, media-friendly and TCP-friendly congestion control algorithms are all based on TFRC and fall into two categories: *multiplicative* and *additive*. Both types consider a media-friendly function $\alpha(q(t))$ which embodies the usefulness of increasing TFRC's throughput passed the rate computed with equation (1) to the streaming application and $q(t)$ is a n-dimensional function giving the values of various media characteristics across time. For simplicity, from now on we will discard the media characteristics $q(t)$ from our notation and write simply $\alpha(t)$.

Multiplicative media-friendly congestion control algorithms use a formula like the following to compute the transmission rate [8]:

$$X(t) = \alpha(t) * X_{TFRC}(t) \quad (2)$$

where $X_{TFRC}(t)$ is given by equation (1). In this type of congestion controls the transmission rate computed by TFRC is altered in a multiplicative way.

Additive media-friendly congestion control algorithms use a formula like the following to compute the transmission rate [7]:

$$X(t) = X_{TFRC}(t) + \alpha(t) \quad (3)$$

where $X_{TFRC}(t)$ is given by equation (1). In this type of congestion controls the transmission rate computed by TFRC is altered in an additive way by the media-friendly function $\alpha(t)$. The media-friendly function, $\alpha(t)$ may include network characteristics (like the loss rate) as parameters, but it does not include the throughput computed by TFRC, $X_{TFRC}(t)$.

It is important that media-friendly congestion control algorithms are also TCP-friendly. However, all the media-friendly congestion control algorithms we are aware of [7, 8] and all TCP-friendly algorithms view the *TCP-friendliness* characteristic as a long term characteristic. We argue that it is important to distinguish between two types of TCP-friendliness: *long-term TCP-friendliness* describing the bandwidth usage and relation to other flows during the duration of the whole streaming session and *short-term TCP-friendliness* describing the local impact on other flows in a small period of time. The short-term TCP-friendliness can be important for flows with a short life time (e.g. web connections). For a short lifespan flow, it is not fair if a flow that is TCP-friendly on the long term consumes twice as much bandwidth as the short lived flow, during its short existence. Of course, by its very nature, a media-friendly flow occasionally consumes on short timescales more bandwidth than a TCP-friendly flow. This slight unfairness is inevitable in any media-friendly congestion control algorithm. It is even present in TFRC and also in TCP. On short time scales, 2 TCP flows get different throughputs even if they share the same network conditions (i.e. round-trip time, loss rate). The idea is to limit this "short timescale unfairness" so that other flows (especially those with a short lifespan) are not affected too much.

At this point we can define the two concepts of TCP-friendliness.

Definition 1. The definition of TCP-friendliness on the long term is the definition of the original TCP-friendliness concept from [1] : A flow is termed *TCP-friendly on the long term* if its long-run average transmission rate, $X(t)$, does not exceed the transmission rate of a TCP flow in the same network conditions.

Definition 2. A flow is termed *TCP-friendly on the short term* if during any 8 loss events time interval (i.e. the time interval spanning over 8 consecutive loss events) the flow does not exceed twice the transmission rate of a TCP flow during the same time interval and in the same network conditions.

4. Simulation study of UTFRC's TCP-friendliness on the short term

Simulation results will be presented in the extended version of this paper.

5. Conclusions

In this paper we presented a classification of media-friendly congestion control algorithms and we underlined the need for the concept of TCP-friendliness on the short term. Consequently we introduced the concept of *TCP-friendliness on the short term* and explained its usefulness for multimedia flows. Simulations characterizing the TCP-friendliness on the short term of a media-friendly congestion control algorithm, UTFRC[8], will be presented in the extended version of this paper.

References

[1] S. Floyd, K. Fall, *Promoting the Use of End-to-End Congestion Control in the Internet*, IEEE/ACM Transactions on Networking, 7(4), pp. 458-472, Aug. 1999.

[2] S. Floyd, M. Handley, J. Padhye, J. Widmer, *Equation-Based Congestion Control for Unicast Applications*, ACM SIGCOMM 2000.

[3] R. Rejaie, M. Handley, and D. Estrin, *An End-to-end Rate-based Congestion Control Mechanism for Realtime Streams in the Internet*, In Proceedings of INFOCOMM 99, 1999.

[4] D. Sisalem and H. Schulzrinne, *The Loss-Delay Based Adjustment Algorithm: A TCP-Friendly Adaption Scheme*, In Proceedings of NOSSDAV98, 1998.

[5] I. Rhee, V. Ozdemir, and Y. Yi, *TEAR: TCP Emulation at Receivers Flow Control for Multimedia Streaming*, April 2000. NCSU Technical Report.

[6] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, *Modeling TCP Throughput: A Simple Model and its Empirical Validation*, SIGCOMM Symposium on Communications Architectures and Protocols, Aug. 1998.

[7] J. Yan, K. Katrinis, M. May, B. Plattner, *Media- and TCP-Friendly Congestion Control for Scalable Video Streams*, IEEE Transactions on Multimedia, Vol. 8, No. 2, April, 2006.

[8] A. Sterca, *UTFRC - Utility-driven TCP-Friendly Rate Control for Multimedia Streams*, Proc. of the 17th Euromicro International Conference on Parallel, Distributed and Network-Based Processing, IEEE Computer Society, Germany, February 2009.

[9] D. Bansal, H. Balakrishnan, S. Floyd, and S. Shenker, *Dynamic behavior of slowly-responsive congestion control algorithms*, In Proc. of ACM SIGCOMM01, San Diego, California, USA, August 2001.

[10] Y. Richard Yang, M. Sik Kim, and Simon S. Lam, *Transient Behaviors of TCP-friendly Congestion Control Protocols*, In Proc. of IEEE Infocom2001, March 2001.

[11] S. Floyd, M. Handley and J. Padhye, *A Comparison of Equation-Based and AIMD Congestion Control*, ACIRI, February 2000, http://www.aciri.org/tfrc/.
[12] I. Rhee, L. Xu, *Limitations of Equation-based Congestion Control*, In Proc. of ACM SIGCOMM'05, Philadelphia, Pennsylvania, USA, August 2005.
[13] Z. Wang, S. Banerjee and S. Jamin, *Media-Friendliness of A Slowly-Responsive Congestion Control Protocol*, In Proceedings of NOSSDAV'04, 2004.

Babes-Bolyai University, Faculty of Mathematics and Computer Science, 1, M. Kogalniceanu St., 400084-Cluj-Napoca, Romania
E-mail address: `forest@cs.ubbcluj.ro`

E-mail address: `vancea@cs.ubbcluj.ro`