Network Working Group
Internet Draft
Intended Status: Informational
Expires: April 17, 2013
X. Zhu
R. Pan
Cisco Systems
October 14, 2012

NADA: A Unified Congestion Control Scheme for Real-Time Media
draft-zhu-rmcat-nada-00

Abstract

This document describes a scheme named network-assisted dynamic adaptation (NADA), a novel congestion control approach for interactive real-time media applications, such as video conferencing. In the proposed scheme, the sender regulates its sending rate based on either implicit or explicit congestion signaling, in a unified approach. The scheme can reap the benefits of explicit congestion notification markings from network nodes. It also maintains consistent sender behavior in the absence of such markings, by reacting to queuing delays instead.

We present here the overall system architecture, recommended behaviors at the sender and the receiver, as well as expected network nodes operations. Results from extensive simulation studies of the proposed scheme are available upon request.

Table of Contents

1. Introduction

Interactive real-time media applications bring about a unique set of challenges for congestion control. Unlike TCP, the mechanism used for real-time media needs to adapt fast to instantaneous bandwidth changes, accommodate fluctuations in the output of video encoder rate control, and cause low queuing delay over the network. An ideal scheme should also make effective use of all types of congestion signals, including packet losses, queuing delay, and explicit congestion notification (ECN) markings.

Based on the above considerations, we present a scheme named network-assisted dynamic adaptation (NADA). The proposed design benefits from explicit congestion control signals (e.g., ECN markings) from the network, and remains compatible in the presence of implicit signals (delay or loss) only. In addition, it supports weighted bandwidth sharing among competing video flows.

This documentation describes the overall system architecture, recommended designs at the sender and receiver, as well as expected network nodes operations. The signaling mechanism consists of standard RTP timestamp [RFC3550] and standard RTCP feedback reports.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. System Model

The system consists of the following elements:

* Incoming media stream, in the form of consecutive raw video frames and audio samples;

* Media encoder with rate control capabilities. It takes the incoming media stream and encodes it to an RTP stream at a target bit rate R_o. Note that the actual output rate from the encoder R_v may fluctuate randomly around R_o. Also, the encoder can only change its rate at rather coarse time intervals, on the order of seconds.

* RTP sender, responsible for calculating the target bit rate R_o based on network congestion signals (delay or ECN marking reports from the receiver), and for regulating the actual sending rate R_s accordingly. The difference between the video

encoder output R_v and sending rate R_s are absorbed in a rate shaping buffer. The buffer size L_s, together with R_v, influences the calculation of R_s. The RTP sender also generates RTP timestamp.

* RTP receiver, responsible for measuring and estimating end-to-end queuing delay d based on sender RTP timestamp. In the presence of ECN markings, it also maintains the statistics of the marking ratio p. The receiver feeds these statistics back to the sender via periodic RTCP reports.

* Network node, with several modes of operation. The system can work with the default behavior of a simple drop tail queue. It can also benefit from advanced AQM features such as RED-based ECN marking, and PCN marking using a token bucket algorithm.

In the following, we will elaborate on the respective operations at the network node, the sender, and the receiver.

4. Network Node Operations

We consider three variations of queue management behavior at the network node, leading to either implicit or explicit congestion signals.

4.1 Default behavior of drop tail

In conventional network with drop tail or RED queues, congestion is inferred from the estimation of end-to-end queuing delay. No special action is required at network node.

This leads to the default operation of delay-based congestion control at the sender.

4.2 ECN marking

In this mode, the network node randomly marks the ECN field in the IP packet header following the Random Early Detection (RED) algorithm [RFC2309]. Calculation of the marking probability involves the following steps:

* upon packet arrival, update smoothed queue size q_avg as:

$$q\_avg = alpha*q + (1-alpha)*q\_avg.$$

The smoothing parameter alpha is a value between 0 and 1. A value of alpha=0 corresponds to performing no smoothing at all, and calculating the marking probability based on instantaneous queue size.

* calculate marking probability p as:

$$p = 0, \text{ if } q<q_{lo};$$

$$p = p_{max}*\frac{q_{avg} - q_{lo}}{q_{hi} - q_{lo}}, \text{ if } q_{lo}<=q<q_{hi};$$

$$p = 1, \text{ if } q>=q_{hi}.$$

Here, q_{lo} and q_{hi} corresponds to the low and high thresholds of queue occupancy. The maximum parking probability is p_{max}.

The ECN markings will trigger the ECN-enabled mode of sender behavior.

4.3 PCN marking

As a more advanced feature, we also envision network nodes which support PCN marking based on virtual queues. In such a case, the marking probability of the ECN bit in the IP packet header is calculated as follows:

* upon packet arrival, meter packet against token bucket (r,b);

* update token level b_{tk};

* calculate the marking probability as:

$$p = 0, \text{ if } b-b_{tk} < b_{lo};$$

$$p = p_{max}* \frac{b-b_{tk}-b_{lo}}{b_{hi}-b_{lo}}, \text{ if } b_{lo}<= b-b_{tk} <b_{hi};$$

$$p = 1, \text{ if } b-b_{tk}>=b_{hi}.$$

Here, the token bucket lower and upper limits are denoted by b_{lo} and b_{hi}, respectively. The parameter b indicates the size of the token bucket. The parameter r is chosen as $r=gamma*C$, where $gamma<1$ is the target utilization ratio and C designates link capacity. The maximum marking probability is p_{max}.

Note that the sender will respond to the observation of markings with exactly the same ECN-enabled mode as in last section. However, the virtual queuing mechanism at the network from the PCN marking algorithm will lead to additional benefits such as zero standing queues and smoother streaming rate.

4.4 Comments and Discussions

In all three flavors described above, the network queue operates with the simple first-in-first-out (FIFO) principle. There is no need to maintain per-flow state. Such a simple design ensures that the system can scale easily with large number of video flows and high link capacity.

5. Sender Behavior

As illustrated in Fig. 1, the sender comprises four modules: a) encoder rate control; b) rate shaping buffer; c) encoder target rate calculator, and d) sending rate calculator.

The following sections describe these modules in further details, and explain how they interact with each other.

```
------------
|          |        Rv       --------------
|          |---------------->     | | | | | -------------->
|  Encoder |                      | | | | |        / \
|          |                 --------------         |
------------              Rate Shaping Buffer       |
     |                               |              |
     |                               |              |
     |                               |              | Rs
     | Ro                            |              |
     |                          Ls   |              |
-----------------                    |      -----------------
|               |                    |      |               |
|  Target Rate  |                    |----->| Sending Rate  |
|  Calculation  |------------------------> | Calculation   |
|               |                           -----------------
----------------
     /\
      |
      --------------- RTCP report (loss/delay/ECN/PC)
```

Figure 1 Sender Structure

5.1 Encoder rate control

The encoder rate control procedure has the following characteristics:

* Rate changes can happen only at large intervals, on the order of seconds.

* Given a target rate R_o, the encoder output rate may randomly fluctuate around it.

* The encoder output rate is further constrained by video content complexity. The range of the final rate output is [R_min, R_max]. Note that it's content-dependent, and may change over time.

5.2 Rate shaping buffer

A rate shaping buffer is employed at the sender, to absorb any instantaneous mismatch between encoder rate output R_v and regulated sending rate R_s. The size of the buffer evolves over time, as:

```
L_s(t) = max [0, L_s(t-tau)+R_v*tau-R_s*tau].
```

A large rate shaping buffer contributes to higher end-to-end delay, which may harm the performance of real-time media communications. Therefore, the sender has a strong incentive to constrain the size of the shaping buffer. It can either deplete it faster by choosing a larger sending rate R_s, or limit its growth by reducing the video encoder target rate R_o.

5.3 Encoder target rate calculator

The sender calculates the encoder target rate based on network congestion information from receiver RTCP reports. When only delay information is available, the target rate is calculated as

```
                         R_max-R_min
R_o = R_min + kappa*w*-------------      (1)
                              d
```

Here, R_min and R_max denote the content-dependent rate range the encoder can produce. The weight of priority level is w. The scaling factor kappa can be tuned to determine how sensitive the rate adaptation scheme is in reaction to fluctuations in observed delay d. The final target rate Ro is clipped within the range of [Rmin, Rmax].

5.3.1 Slow-start behavior

In addition, the initial sending rate of a stream is regulated to grow linearly, no more than R_ss at time t:

```
                    t-t_0
R_ss(t) = R_min + -------(R_max-R_min).
```

```
                              T
```

The start time of the stream is t_0, and T represents the time horizon over which the slow-start mechanism is effective. The encoder target rate is chosen to be the minimum of R_o and R_ss in the first T seconds.

5.3.2 ECN-enabled mode

If the receiver reports on observed ECN marking probability p, the target rate calculation of Eq. (1) is replaced by the following, with a scaling factor eta:

```
                             R_max - R_min
        R_o = R_min + eta*w*----------------.        (2)
                                 p
```

All other procedures remain the same.

Note that the sender does not need to differentiate whether the network node operates with RED-based ECN marking, or token-bucket-level-based PCN marking. It reacts to observed ECN marking probabilities in exactly the same manner.

5.4 Sending rate calculator

Finally, the actual outgoing rate over the network is R_s. Its value is calculated based on both the encoder target rate and rate shaping buffer size, as follows:

```
                             L_s
        R_s = R_o + beta * -------.
                            tau_v
```

The first term indicates the rate calculated from network congestion feedback alone. The second term exerts additional pressure to send out more packets, if the rate shaping buffer is building up. The scaling factor beta can be tuned to balance between these two competing goals.

6. Receiver Behavior

The role of the receiver is fairly straightforward. It observes and estimates end-to-end queuing delay d and ECN marking ratio p of the stream. The former can be obtained from the RTP timestamp provided by the sender. The latter can be obtained by keeping a running average of the number of marked and un-marked packets. The detailed mechanisms for obtaining such estimates can be varied, and is out of the scope of this paper.

Periodically, the receiver sends back the updated values of d and p in RTCP messages, to aid the sender in its calculation of target rate. Note that the size of acknowledgement packets are typically on the order of tens of bytes, and are significantly smaller than average video packet sizes. Therefore, the bandwidth overhead of the receiver acknowledgement stream is sufficiently low.

7. Incremental Deployment

One nice property of proposed design is the consistent behavior of video end points regardless of variations in network node operations. This facilitates gradual, incremental adoption of the scheme.

To start off with, the scheme operating in delay-assisted congestion control mode can be implemented without any explicit support from the network.

When ECN is enabled at the network nodes, together with RED-based marking, the sender can react to these explicit congestion signals instead. Ultimately, networks equipped with proactive marking based on token bucket level metering can reap the additional benefits, including zero standing queues and more smooth streaming rates.

8. IANA Considerations

There are no actions for IANA.

9. References

9.1 Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

9.2 Informative References

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, September 2001.

[RFC2309] Braden, B., Clark, D., Crowcroft, J., Davie, B., Deering, S., Estrin, D., Floyd, S., Jacobson, V., Minshall, G., Partridge, C., Peterson, L., Ramakrishnan, K., Shenker,

S., Wroclawski, J., and L. Zhang, "Recommendations on Queue Management and Congestion Avoidance in the Internet", RFC 2309, April 1998.

Authors' Addresses

Xiaoqing Zhu
Cisco Systems,
510 McCarthy Blvd,
Milpitas, CA 95134, USA
EMail: xiaoqzhu@cisco.com

Rong Pan
Cisco Systems
510 McCarthy Blvd,
Milpitas, CA 95134, USA
Email: ropan@cisco.com