Network Working Group
Internet-Draft
Intended status: Informational
Expires: December 19, 2014

Z. Sarker
Ericsson AB
V. Singh
Aalto University
X. Zhu
M. Ramalho
Cisco Systems

June 17, 2014

Test Cases for Evaluating RMCAT Proposals
draft-sarker-rmcat-eval-test-01

Abstract

The Real-time Transport Protocol (RTP) is used to transmit media in multimedia telephony applications, these applications are typically required to implement congestion control. The RMCAT working group is currently working on candidate algorithms for such interactive real-time multimedia applications. This document describes the test cases to be used in the performance evaluation of those candidate algorithms.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

This memo describes a set of test cases for evaluating candidate RMCAT congestion control algorithm proposals, it is based on the guidelines enumerated in [I-D.ietf-rmcat-eval-criteria] and the requirements discussed in [I-D.ietf-rmcat-cc-requirements]. The test cases cover basic usage scenarios and are described using a common structure, which allows for additional test cases to be added to those described herein to accommodate other topologies and/or the modeling of different path characteristics. It is the intention of this work to capture the consensus of the RMCAT working group participants regarding the test cases upon which the performance of the candidate RMCAT proposals should be evaluated.

2. Terminology

The terminology defined in RTP [RFC3550], RTP Profile for Audio and Video Conferences with Minimal Control [RFC3551], RTCP Extended Report (XR) [RFC3611], Extended RTP Profile for RTCP-based Feedback (RTP/AVPF) [RFC4585], Support for Reduced-Size RTCP [RFC5506], and RTP Circuit Breaker algorithm [I-D.ietf-avtcore-rtp-circuit-breakers] apply.

3. Structure of Test cases

All test cases in this document follow a basic structure allowing implementers to describe a new test scenario without repeatedly explaining common attributes. The structure includes a general description section that describes the test case and its motivation. Additionally the test case defines a set of attributes that characterize the testbed, i.e., the network path between communicating peers and the diverse traffic sources.

- Define the test case:

  * General description: describes the motivation and the goals of the test case.

  * Expected behavior: describe the desired rate adaptation behaviour.

  * Define a check-list to evaluate the desired behaviour: this indicates the minimum set of metrics (e.g., link utilization, media sending rate) that a proposed algorithm needs to measure to validate the expected rate adaptation behaviour. It should also indicate the time granularity (e.g., averaged over 10ms, 100ms, or 1s) for measuring certain metrics. Typical measurement interval is 200ms.

o Define testbed topology: every test case needs to define an evaluation testbed topology. Figure 1 shows such an evaluation topology. In this evaluation topology, S1..Sn are traffic sources. These sources generate media traffic and use either an RMCAT candidate congestion control algorithm or other congestion control algorithm designed for media, such as TFRC. R1..Rn are the corresponding receivers. A test case can have one or more such traffic sources (S) and corresponding receivers (R). The path from the source to destination is denoted as forward and the path from a destination to a source is denoted as backward. The following basic structure of test case has been described from the perspective of media generating endpoints attached on the left-hand side of Figure 1. In this setup, media flows in forward direction and corresponding feedback/control messages flow in the backward direction. However, it is also possible to set up the test with media flowing in both forward and backward directions. In that case, unless otherwise specified by the test case, it is expected that the backward path does not introduce any congestion related impairments and has enough capacity to accommodate both media and feedback/control messages. It should be noted that depending on the test cases it is possible to have different path characteristics in of the either directions.

o

```
   +---+                                                           +---+
   |S1 |====== \                 Forward -->              / =======|R1 |
   +---+       \\                                        //        +---+
                \\                                      //
   +---+       +------+                              +------+      +---+
   |S2 |=======|  A   |----------------------------->|  B   |=======|R2 |
   +---+       |      |<-----------------------------|      |      +---+
               +------+                              +------+
   (...)         //                                     \\         (...)
                //          <-- Backward                 \\
   +---+       //                                         \\       +---+
   |Sn |====== /                                           \ ======|Rn |
   +---+                                                           +---+
```

Figure 1: Example of A Testbed Topology

In a laboratory testbed environment there may exist a significant amount of traffic on portions of the network path between the endpoints that is not desired for the purposes of these RMCAT tests. Some of this traffic may be generated by other processes on the endpoints themselves (e.g., discovery protocols) or by other endpoints not presently under test. It is recommended not to route traffic generated by endpoints that are not under test through the test bed. Additionally, it is recommended to route non-RMCAT traffic generated by the endpoints under test around the bottleneck links specified herein.

- Define testbed attributes:

  * Duration: defines the duration of the test.

  * Path characteristics: defines the end-to-end transport level path characteristics of the testbed in a particular test case. Two sets of attributes describe the path characteristics, one for the forward path and the other for the backward path. The path characteristics for a particular path direction is applicable to all the Sources "S" sending traffic on that path. If only one attribute is specified, it is used for both path directions, however, unless specified the reverse path has no capacity restrictions and no path loss.

    + Path direction: forward or backward.

    + Bottleneck-link capacity: defines minimum capacity of the end-to-end path

    + One-way propagation delay: describes the end-to-end latency along the path when network queues are empty, i.e., the time it takes for a packet to go from the sender to the receiver without encountering any queuing delay.

    + Maximum end-to-end jitter: defines the maximum jitter that can be observed along the path.

    + Bottleneck queue type: for example, Droptail, FQ-CoDel, or PIE.

    + Bottleneck queue size: defines size of queue in terms of queuing time when the queue is full (in milliseconds).

    + Path loss ratio: characterizes the non-congested, additive, losses to be generated on the end-to-end path. MUST describe the loss pattern or loss model used to generate the losses.

  * Application-related: defines the traffic source behaviour for implementing the test case

    + Media traffic Source: defines the characteristics of the media sources. When using more than one media source, the different attributes are enumerated separately for each different media source.

- Media type: Video/Voice/Application/Text

- Media flow direction: forward, backward or both.

- Number of media sources: defines the total number of media sources

- Media codec: Constant Bit Rate (CBR) or Variable Bit Rate (VBR)

- Media source behaviour: describes the media encoder behavior. It defines the main parameters that affect the adaptation behaviour. This may include but not limited to:

    o Adaptability: describes the adaptation options. For example, in the case of video it defines the following ranges of adaptation: bit rate, frame rate, video resolution. Similarly, in the case of voice, it defines the range of bit rate adaptation, the sampling rate variation, and the variation in packetization interval.

    o Output variation : for a VBR encoder it defines the encoder output variation from the average target rate over a particular measurement interval. For example, on average the encoder output may vary between 5% to 15% above or below the average target bit rate when measured over a 100 ms time window. The time interval over which the variation is specified must be provided.

    o Responsiveness to a new bit rate request: the lag in time between a new bit rate request and actual rate changes in encoder output. Depending on the encoder, this value may be specified in absolute time (e.g. 10ms to 1000ms) or other appropriate metric (next frame interval time).

- Media content: describes the chosen media sequences; For example, test sequences are available at: [xiph-seq] and [HEVC-seq].

- Media timeline: describes the point when the media source is introduced and removed from the testbed. For example, the media source may start transmitting immediately when the test case begins, or after a few seconds.

- Startup behaviour: the media starts at a defined bit rate, which may be the minimum, maximum bit rate, or a value in between (in Kbps).

+ Competing traffic source: describes the characteristics of the competing traffic source, the different types of competing flows are enumerated in [I-D.ietf-rmcat-eval-criteria].

  - Traffic direction: forward, backward or both.

  - Type of sources: defines the types of competing traffic sources. Types of competing traffic flows are listed in [I-D.ietf-rmcat-eval-criteria]. For example, the number of TCP flows connected to a web browser, the mean size and distribution of the content downloaded.

  - Number of sources: defines the total number of competing sources of each media type.

  - Congestion control: enumerates the congestion control used by each type of competing traffic.

  - Traffic timeline: describes when the competing traffic starts and ends in the test case.

* Additional attributes: describes attributes essential for implementing a test case which are not included in the above structure. These attributes MUST be well defined, so that other implementers are able to implement it.

Any attribute can have a set of values (enclosed within "[]"). Each member value of such a set MUST be treated as different value for the same attribute. It is desired to run separate tests for each such attribute value.

The test cases described in this document follow the above structure.

4. Recommended Evaluation Settings

This section describes recommended test case settings and could be overwritten by the respective test cases.

4.1. Evaluation metircs

To evaluate the performance of the candidate algorithms it is expected to log enough information to visualize the following metrics at a fine enough time granularity:

1. Flow level:

    A. End-to-end delay for the RMCAT flow.

    B. Variation in sending bit rate and goodput. Mainly observing the frequency and magnitude of oscillations.

    C. Packet losses observed at the receiving endpoint

    D. Feedback message overhead

    E. Convergence time.

2. Transport level:

    A. Bandwidth utilization

    B. Queue length (milliseconds at specified path capacity):

        + average over the length of the session

        + 5 and 95 percentile

        + median, maximum, minimum

4.2. Path characteristics

Each path between a sender and receiver as described in Figure 1 have the following characteristics unless otherwise specified in the test case.

- o Path direction: forward and backward.

- o Bottleneck-link capacity: 4Mbps.

- o One-Way propagation delay: 50ms. It is encouraged to test with additional propagation delays mentioned in Appendix A.1

- o Maximum end-to-end jitter: 30ms. Jitter models are described in Appendix B.1

- Bottleneck queue type: Drop tail. It is encouraged to test with other AQM schemes, such as FQ-CoDel and PIE.

- Bottleneck queue size: 300ms.

- Path loss ratio: 0%.

Examples of additional network parameters are discussed in Appendix A.

4.3. Media source

Unless otherwise specified, each test case will include one or more media sources as described below.

- Media type: Video

  * Media codec: VBR

  * Media source behaviour:

    + Adaptability:

      - Bit rate range: 150 Kbps - 1.5 Mbps. In real-life applications the bitrate range can vary a lot depending on the provided service, for example, the maximum bitrate can be up to 4Mbps. However, for running tests to evaluate the congestion control algorithms it is more important to have a look at how they are reacting to certain amount of bandwidth change. Also it is possible that the media traffic generator used in a particular simulator or testbed if not capable of generating higher bitrate. Hence we have selected a suitable bitrate range typical of consumer-grade video conferencing applications in designing the test case. If a different bitrate range is used in the test cases, the end-to-end path capacity values will also need to be scaled accordingly.

      - Frame resolution: 144p - 720p (or 1080p)

      - Frame rate: 10fps - 30fps

    + Variation from target bitrate: +/-5%. Unless otherwise specified in the test case, bitrate variation SHOULD be calculated over one (1) second period of time.

    + Responsiveness to new bit rate request: 100ms

* Media content: The media content should represent a typical video conversational scenario with head and shoulder movement. We recommend to use Foreman video sequence.

* Media startup behaviour: 150Kbps. It should be noted that applications can use smart ways to select an optimal startup bitrate values for a certain network condition. In such cases the candidate proposals MAY show the effectiveness of such smart approach as an additional information for the evaluation process.

o Media type: Audio

* Media codec: CBR

* Media bitrate: 20Kbps

5. Basic Test Cases

5.1. Variable Available Capacity with Single RMCAT flow

In this test case the bottleneck-link capacity between the two endpoints varies over time. This test is designed to measure the responsiveness of the candidate algorithm. This test tries to address the requirements in [I-D.ietf-rmcat-cc-requirements], which requires the algorithm to adapt the flow(s) and provide lower end-to-end latency when there exists:

o an intermediate bottleneck

o change in available capacity (e.g., due to interface change, routing change).

o maximum Media Bit Rate is Greater than Link Capacity. In this case, the application will attempt to ramp up to its maximum bit rate, since the link capacity is limited to a value lower, the congestion control scheme is expected to stabilize the sending bit rate close to the available bottleneck capacity. This situation can occur when the endpoints are connected via thin long networks even though the advertised capacity of the access network may be higher.

It should be noted that the exact variation in available capacity due to any of the above depends on the under-lying technologies. Hence, we describe a set of known factors, which may be extended to devise a more specific test case targeting certain behaviour in a certain network environment.

Expected behavior: the candidate algorithm is expected to detect the path capacity constraint, converges to bottleneck link's capacity and adapt the flow to avoid unwanted oscillation when the sending bit rate is approaching the bottleneck link's capacity. The oscillations occur when the media flow(s) attempts to reach its maximum bit rate, overshoots the usage of the available bottleneck capacity, to rectify it reduces the bit rate and starts to ramp up again.

Testbed topology: One media source S1 is connected to corresponding R1. The media traffic is transported over the forward path and corresponding feedback/control traffic is transported over the backward path.

```
                           Forward -->
+---+       +-----+                               +-----+       +---+
|S1 |=======|  A  |------------------------------>|  B  |=======|R1 |
+---+       |     |<------------------------------|     |       +---+
            +-----+                               +-----+
                        <-- Backward
```

Figure 2: Testbed Topology for Limited Link Capacity

To evaluate the performance of the candidate algorithms it is expected to log enough information to visualize the metrics described in Section 4.1 at a fine enough time granularity.

Testbed attributes:

- o Test duration: 100s
- o Path characteristics: as described in Section 4.2
- o Application-related:
    - * Media Traffic:
        - + Media type: Video
            - - Media direction: forward.
            - - Number of media sources: One (1)
            - - Media timeline:
                - o Start time: 0s.
                - o End time: 99s.

+ Media type: Audio

  - Media direction: forward.

  - Number of media sources: One (1)

  - Media timeline:

    o Start time: 0s.

    o End time: 99s.

* Competing traffic:

  + Number of sources : Zero (0)

o Test Specific Information:

  * This test uses the following one way propagation delays of 50 ms and 100 ms.

  * This test uses bottleneck path capacity variation as listed in Table 1

Variation pattern index	Path direction	Start time	Path capacity
One	Forward	0s	1Mbps
Two	Forward	40s	2.5Mbps
Three	Forward	60s	600Kbps
Four	Forward	80s	1Mbps

Table 1: Path capacity variation pattern for forward direction

5.2. Variable Available Capacity with Multiple RMCAT flows

This test case is similar to Section 5.1. However in addition this test will also consider persistent network load due to competing traffic.

Expected behavior: the candidate algorithms is expected to detect the variation in available capacity and adapt the media stream(s) accordingly. The flows stabilize around their maximum bitrate as the as the maximum link capacity is large enough to accommodate the flows. When the available capacity drops, the flow(s) adapts by decreasing its sending bit rate, and when congestion disappears, the flow(s) are again expected to ramp up.

To evaluate the performance of the candidate algorithms it is expected to log enough information to visualize the metrics described in Section 4.1 at a fine enough time granularity:

Testbed Topology: Two (2) media sources S1 and S2 are connected to their corresponding destinations R1 and R2. The media traffic is transported over the forward path and corresponding feedback/control traffic is transported over the backward path.

```
+---+                                                       +---+
|S1 |===== \                                      / =======|R1 |
+---+      \\             Forward -->            //        +---+
            \\                                  //
            +-----+                            +-----+
            |  A  |------------------------------>|  B  |
            |     |<------------------------------|     |
            +-----+                            +-----+
              //                                 \\
             //          <-- Backward             \\
+---+       //                                     \\       +---+
|S2 |====== /                                        \ ======|R2 |
+---+                                                        +---+
```

Figure 3: Testbed Topology for Variable Available Capacity

Testbed attributes:

Testbed attributes are similar as described in Section 5.1 except the test specific capacity variation setup.

Test Specific Information: This test uses path capacity variation as listed in Table 2 with a corresponding end time of 125 seconds.

Variation pattern index	Path direction	Start time	Path capacity
One	Forward	0s	4Mbps
Two	Forward	25s	2Mbps
Three	Forward	50s	3.5Mbps
Four	Forward	75s	1Mbps
Five	Forward	100s	2Mbps

Table 2: Path capacity variation pattern for forward direction

5.3. Congested Feedback Link with Bi-directional RMCAT flows

RMCAT WG has been chartered to define algorithms for RTP hence it is assumed that RTCP, RTP header extension or such would be used by the congestion control algorithm in the backchannel. Due to asymmetric nature of the link between communicating peers it is possible for a participating peer to not receive such feedback information due to an impaired or congested backchannel (even when the forward channel might not be impaired). This test case is designed to observe the candidate congestion control behaviour in such an event.

It is expected that the candidate algorithms is able to cope with the lack of feedback information and adapt to minimize the performance degradation of media flows in the forward channel.

It should be noted that for this test case: logs are compared with the reference case, i.e, when the backward channel has no impairments

To evaluate the performance of the candidate algorithms it is expected to log enough information to visualize the metrics described in Section 4.1 at a fine-grained time intervals:

Testbed topology: One (1) media source S1 is connected to corresponding R1, but both endpoints are additionally receiving and sending data, respectively. The media traffic (S1->R1) is transported over the forward path and corresponding feedback/control traffic is transported over the backward path. Likewise media traffic (S2->R2) is transported over the backward path and corresponding feedback/control traffic is transported over the forward path.

```
     +---+                                                         +---+
     |S1 |===== \                Forward -->              / =======|R1 |
     +---+      \\                                       //        +---+
                 \\                                     //
              +-----+                               +-----+
              |  A  |------------------------------>|  B  |
              |     |<------------------------------|     |
              +-----+                               +-----+
                 //                                     \\
                //           <-- Backward                \\
    +---+      //                                         \\       +---+
    |R2 |===== /                                           \ ======|S2 |
    +---+                                                          +---+
```

Figure 4: Testbed Topology for Congested Feedback Link

Testbed attributes:

o Test duration: 100s

o Path characteristics:

  * Bottleneck-link capacity: 2Mbps.

o Application-related:

  * Media Source:

    + Media type: Video

      - Media direction: forward and backward

      - Number of media sources: Two (2)

      - Media timeline:

        o Start time: 0s.

        o End time: 99s.

    + Media type: Audio

      - Media direction: forward and backward

      - Number of media sources: Two (2)

      - Media timeline:

        o Start time: 0s.

        o End time: 99s.

  * Competing traffic:

    + Number of sources : Zero (0)

o Test Specific Information: This test uses path capacity variations to create congested feedback link. Table 3 lists the variation patterns applied to the forward path and Table 4 lists the variation patterns applied to the backward path.

Variation pattern index	Path direction	Start time	Path capacity
One	Forward	0s	2Mbps
Two	Forward	20s	1Mbps
Three	Forward	40s	500Kbps
Four	Forward	60s	2Mbps

Table 3: Path capacity variation pattern for forward direction

Variation pattern index	Path direction	Start time	Path Capacity
One	Backward	0s	2Mbps
Two	Backward	35s	800Kbps
Three	Backward	70s	2Mbps

Table 4: Path capacity variation pattern for backward direction

5.4. Competing Flows with Same RMCAT Algorithm

In this test case, more than one RMCAT media flow shares the bottleneck link and each of them uses the same congestion control algorithm. This is a typical scenario where a real-time interactive application sends more than one media flows to the same destination and these flows are multiplexed over the same port. In such a scenario it is likely that the flows will be routed via the same path and need to share the available bandwidth amongst themselves. For the sake of simplicity it is assumed that there are no other non-RMCAT competing traffic sources in the bottleneck link and that there is sufficient capacity to accommodate all the flows individually. While this appears to be a variant of the test case defined in Section 5.2, it focuses on the capacity sharing aspect of the candidate algorithm. The previous test case, on the other hand, measures adaptability, stability, and responsiveness of the candidate algorithm.

Expected behavior: It is expected that the competing flows will converge to an optimum bit rate to accommodate all the flows with minimum possible latency and loss. Specifically, the test introduces three media flows at different time instances, when the second flow appears there should still be room to accommodate another flow on the bottleneck link. Lastly, when the third flow appears the bottleneck link should be saturated.

To evaluate the performance of the candidate algorithms it is expected to log enough information to visualize the metrics described in Section 4.1 at a fine enough time granularity:

Testbed topology: Three media sources S1, S2, S3 are connected to respective R1, R2, R3. The media traffic is transported over the forward path and corresponding feedback/control traffic is transported over the backward path.

```
   +---+                                                         +---+
   |S1 |===== \                Forward -->             / =======|R1 |
   +---+      \\                                       //        +---+
               \\                                     //
   +---+       +-----+                               +-----+       +---+
   |S2 |=======|  A  |------------------------------>|  B  |=======|R2 |
   +---+       |     |<------------------------------|     |       +---+
               +-----+                               +-----+
                 //                                     \\
                //          <-- Backward                 \\
   +---+       //                                         \\       +---+
   |S3 |====== /                                           \ ======|R3 |
   +---+                                                           +---+
```

Figure 5: Testbed Topology for Multiple RMCAT Flows

Testbed attributes:

o Test duration: 120s

o Path characteristics:

   * Bottleneck-link capacity: 3.5Mbps

o Application-related:

   * Media Source:

      + Media type: Video

         - Media direction: forward.

         - Number of media sources: Three (3)

         - Media timeline: New media flows are added sequentially, at short time intervals. See test specific setup below.

      + Media type: Audio

- Media direction: forward.

- Number of media sources: Three (3)

- Media timeline: New media flows are added sequentially, at short time intervals. See test specific setup below.

* Competing traffic:

  + Number of sources : Zero (0)

o Test Specific Information:

* Media flow timeline:

  + Flow ID: One (1)

  + Start time: 0s

  + End time: 119s

* Media flow timeline:

  + Flow ID: Two (2)

  + Start time: 20s

  + End time: 119s

* Media flow timeline:

  + Flow ID: Three (3)

  + Start time: 40s

  + End time: 119s

5.5. Round Trip Time Fairness

In this test case, multiple RMCAT media flows share the bottleneck link, but the end-to-end path latency for each RMCAT flow is different. For the sake of simplicity it is assumed that there are no other non-RMCAT competing traffic sources in the bottleneck link and that there is sufficient capacity to accommodate all the flows. While this appears to be a variant of test case 5.2, it focuses on the capacity sharing aspect of the candidate algorithm under different RTTs.

It is expected that the competing flows will converge to bit rates to accommodate all the flows with minimum possible latency and loss. Specifically, the test introduces five media flows at the same time instance.

To evaluate the performance of the candidate algorithms it is expected to log enough information to visualize the metrics described in Section 4.1 at a fine enough time granularity:

Testbed Topology: Five (5) media sources S1,S2,..,S5 are connected to their corresponding media sinks R1,R2,..,R5. The media traffic is transported over the forward path and corresponding feedback/control traffic is transported over the backward path. The topology is the same as in Section 5.4. The end-to-end path delays are: 10ms for S1-R1, 25ms for S2-R2, 50ms for S3-R3, 100ms for S4-R4, and 150ms S5-R5, respectively.

Testbed attributes:

- o Test duration: 120s

- o Path characteristics:

    * One-Way propagation delay for each flow: 10ms, 25ms, 50ms, 100ms, 150ms.

- o Application-related:

    * Media Source:

        + Media type: Video

            - Media direction: forward

            - Number of media sources: Five (5)

            - Media timeline:

                o Start time: 0s.

                o End time: 119s.

        + Media type: Audio

            - Media direction: forward.

            - Number of media sources: Five (5)

- Media timeline:

  o Start time: 0s.

  o End time: 119s.

* Competing traffic:

  + Number of sources : Zero (0)

o Test Specific Information: None

5.6. RMCAT Flow competing with a long TCP Flow

In this test case, one or more RMCAT media flows share the bottleneck link with at least one long lived TCP flows. Long lived TCP flows download data throughout the session and are expected to have infinite amount of data to send and receive. This is a scenario where a multimedia application co-exists with a large file download. The test case measures the adaptivity of the candidate algorithm to competing traffic. It addresses the requirement 3 in [I-D.ietf-rmcat-cc-requirements].

Expected behavior: depending on the convergence observed in test case 5.1 and 5.2, the candidate algorithm may be able to avoid congestion collapse. In the worst case, the media stream will fall to the minimum media bit rate.

To evaluate the performance of the candidate algorithms it is expected to log enough information to visualize the following metrics in addition to the metrics described in Section 4.1 at a fine enough time granularity:

1. Flow level:

   A. TCP throughput.

Testbed topology: One (1) media source S1 is connected to corresponding media sink, R1. In addition, there is a long-live TCP flow sharing the same bottleneck link. The media traffic is transported over the forward path and corresponding feedback/control traffic is transported over the backward path. The TCP traffic goes over the forward path from, S_tcp with acknowledgement packets flowing along the backward path from, R_tcp.

```
        +--+                                                     +--+
        |S1|===== \              Forward -->            / =======|R1|
        +--+      \\                                   //        +--+
                   \\                                 //
               +-----+                             +-----+
               |  A  |---------------------------->|  B  |
               |     |<----------------------------|     |
               +-----+                             +-----+
                    //                                 \\
                   //        <-- Backward               \\
       +-----+    //                                     \\    +-----+
       |S_tcp|=== /                                        \ ===|R_tcp|
       +-----+                                                 +-----+
```

Figure 6: Testbed Topology for TCP vs RMCAT Flows

Testbed attributes:

o Test duration: 120s

o Path characteristics:

   * Bottleneck-link capacity: 2Mbps

   * Bottleneck queue size: [20ms, 300ms, 1000ms]

o Application-related:

   * Media Source:

      + Media type: Video

         - Media direction: forward

         - Number of media sources: One (1)

         - Media timeline:

            o Start time: 5s.

            o End time: 119s.

      + Media type: Audio

         - Media direction: forward

         - Number of media sources: One (1)

            - Media timeline:

               o Start time: 5s.

               o End time: 119s.

      * Additionally, implementers are encouraged to run the experiment with multiple media sources.

      * Competing traffic:

         + Number and Types of sources : one (1), long-lived TCP

         + Traffic direction : forward

         + Congestion control: Default TCP congestion control.

         + Traffic timeline:

            - Start time: 0s.

            - End time: 119s.

   o Test Specific Information: None

5.7. RMCAT Flow competing with short TCP Flows

In this test case, one or more RMCAT media flow shares the bottleneck link with multiple short-lived TCP flows. Short-lived TCP flows resemble the on/off pattern observed in the web traffic, wherein clients (browsers) connect to a server and download a resource (typically a web page, few images, text files, etc.) using several TCP connections (up to 4). This scenario shows the performance of the multimedia application when several browser windows are active. The test case measures the adaptivity of the candidate algorithm to competing web traffic, it addresses the requirements 1.E in [I-D.ietf-rmcat-cc-requirements].

Depending on the number of short TCP flows, the cross-traffic either appears as a short burst flow or resembles a long TCP flow. The intention of this test is to observe the impact of short-term burst on the behaviour of the candidate algorithm.

To evaluate the performance of the candidate algorithms it is expected to log enough information to visualize the following metrics in addition to the metrics described in Section 4.1 at a fine enough time granularity:

1. Flow level:

   A. Variation in the sending rate of the TCP flow.

   B. TCP throughput.

Testbed topology: The topology described here is same as the one described in Figure 6.

Testbed attributes:

- Test duration: 300s

- Path characteristics:

  * Bottleneck-link capacity: 2.0Mbps

- Application-related:

  * Media Source:

    + Media type: Video

      - Media direction: forward

      - Number of media sources: two (2)

      - Media timeline:

        o Start time: 5s.

        o End time: 299s.

    + Media type: Audio

      - Media direction: forward

      - Number of media sources: two (2)

      - Media timeline:

        o Start time: 5s.

        o End time: 299s.

  * Competing traffic:

+ Number and Types of sources : Ten (10), short-lived TCP flows.

+ Traffic direction : forward

+ Congestion algorithm: Default TCP Congestion control.

+ Traffic timeline: Each short TCP flow is modeled as a sequence of file downloads interleaved with idle periods. See test specific setup. Not all short TCPs start at the same time, 2 start in the ON state while 8 start in an OFF stats. The model for the idle times for the OFF state is discussed in the Short-TCP model.

o Test Specific Information:

* Short-TCP traffic model:

+ File sizes: uniform distribution between 100KB to 1MB

+ Idle period: the duration of the OFF state is derived from an exponential distribution with the mean value of 10 seconds.

5.8. Media Pause and Resume

In this test case, more than one real-time interactive media flows share the link bandwidth and all flows reach to a steady state by utilizing the link capacity in an optimum way. At these stage one of the media flow is paused for a moment. This event will result in more available bandwidth for the rest of the flows and as they are on a shared link. When the paused media flow will resume it would no longer have the same bandwidth share on the link. It has to make it’s way through the other existing flows in the link to achieve a fair share of the link capacity. This test case is important specially for real-time interactive media which consists of more than one media flows and can pause/resume media flow at any point of time during the session. This test case directly addresses the requirement number 5 in [I-D.ietf-rmcat-cc-requirements]. One can think it as a variation of test case defined in Section 5.4. However, it is different as the candidate algorithms can use different strategies to increase its efficiency, for example the fairness, convergence time, reduce oscillation etc, by capitalizing the fact that they have previous information of the link.

To evaluate the performance of the candidate algorithms it is expected to log enough information to visualize the following metrics in addition to the metrics described in Section 4.1 at a fine enough time granularity:

1. Flow level:

   A. Variation in sending bit rate and goodput. Mainly observing the frequency and magnitude of oscillations.

Testbed Topology: Same as test case defined in Section 5.4

Testbed attributes: The general description of the testbed parameters are same as Section 5.4 with changes in the test specific setup as below-

o Other test specific setup:

  * Media flow timeline:

    + Flow ID: One (1)

    + Start time: 0s

    + Flow duration: 119s

    + Pause time: not required

    + Resume time: not required

  * Media flow timeline:

    + Flow ID: Two (2)

    + Start time: 0s

    + Flow duration: 119s

    + Pause time: at 40s

    + Resume time: at 60s

  * Media flow timeline:

    + Flow ID: Three (3)

    + Start time: 0s

    + Flow duration:119s

+ Pause time: not required

+ Resume time: not required

6. Other potential test cases

It has been noticed that there are other interesting test cases besides the basis test cases listed above. In many aspects, these additional test cases can help to further evaluate the candidate algorithm. They are listed as below.

6.1. Explicit Congestion Notification Usage

This test case requires to run all the basic test cases with the availability of Explicit Congestion Notification (ECN) [RFC6679] feature enabled. The goal of this test is to exhibit that the candidate algorithms does not fail when ECN signals are available. With ECN signals enabled the algorithms are expected to perform better than their delay based variants.

6.2. Multiple Bottlenecks

In this test case one RMCAT flow, S1->R2 traverse a path with multiple bottlenecks. As illustrated in Figure 7, the first flow (S1->R1) competes with the second RMCAT flow (S2->R2) over the link between A and B which is close to the sender side; again, that flow (S1->R1) competes with the third RMCAT flow (S3->R3) over the link between C and D which is close to the receiver side. The goal of this test is to ensure that the candidate algorithms work properly in the presence of multiple bottleneck links on the end to end path.

Expected behavior: the candidate algorithm is expected to achieve full utilization at both bottleneck links without starving any of the three RMCAT flows.

```
                               Forward ---->


               +---+          +---+        +---+      +---+
               |S2 |          |R2 |        |S3 |      |R3 |
               +---+          +---+        +---+      +---+
                 |              |            |          |
             |              |            |          |
  +---+       +-----+       +-----+      +-----+     +-----+       +---+
  |S1 |=======|  A  |------>|  B  |----->|  C  |---->|  D  |=======|R1 |
  +---+       |     |<------|     |<-----|     |<----|     |       +---+
              +-----+       +-----+      +-----+     +-----+

                       1st                       2nd
                Bottleneck (A->B)         Bottleneck (C->D)

                             <------ Backward
```

Figure 7: Testbed Topology for Multiple Bottlenecks

Testbed topology: Three media sources S1, S2, and S3 are connected to respective destinations R1, R2, and R3. For all three flows the media traffic is transported over the forward path and corresponding feedback/control traffic is transported over the backward path.

Testbed attributes:

- o Test duration: 120s

- o Path characteristics:

  * Path capacity between A and B = 2Mbps.

  * Path capacity between B and C = 8Mbps.

  * Path capacity between C and D = 1.5Mbps.

  * One-Way propagation delay:

    1. Between S1 and R1 : 100ms

    2. Between S2 and R2: 40ms

    3. Between S3 and R3: 40ms

o Application-related:

  * Media Source:

    + Media type: Video

      - Media direction: Forward

      - Number of media sources: Three (3)

      - Media timeline:

        o Start time: 0s.

        o End time: 119s.

    + Media type: Audio

      - Media direction: Forward

      - Number of media sources: Three (3)

      - Media timeline:

        o Start time: 0s.

        o End time: 119s.

  * Competing traffic:

    + Number of sources : Zero (0)

7. Wireless Access Links

7.1. Cellular Network Specific Test Cases

Additional cellular network specific test cases are define in [I-D.draft-sarker-rmcat-cellular-eval-test-cases]

7.2. Wi-Fi Network Specific Test Cases

TBD

[Editor’s Note: We should encourage people to come up with possible WiFi Network specific test cases]

8. Security Considerations

Security issues have not been discussed in this memo.

9. IANA Considerations

There are no IANA impacts in this memo.

10. Acknowledgements

Much of this document is derived from previous work on congestion control at the IETF.

The content and concepts within this document are a product of the discussion carried out in the Design Team.

11. References

11.1. Normative References

[RFC6679] Westerlund, M., Johansson, I., Perkins, C., O'Hanlon, P., and K. Carlberg, "Explicit Congestion Notification (ECN) for RTP over UDP", RFC 6679, August 2012.

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

[RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, July 2003.

[RFC3611] Friedman, T., Caceres, R., and A. Clark, "RTP Control Protocol Extended Reports (RTCP XR)", RFC 3611, November 2003.

[RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey, "Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585, July 2006.

[RFC5506] Johansson, I. and M. Westerlund, "Support for Reduced-Size Real-Time Transport Control Protocol (RTCP): Opportunities and Consequences", RFC 5506, April 2009.

[I-D.ietf-avtcore-rtp-circuit-breakers]
Perkins, C. and V. Singh, "Multimedia Congestion Control: Circuit Breakers for Unicast RTP Sessions", draft-ietf-avtcore-rtp-circuit-breakers-05 (work in progress), February 2014.

[I-D.ietf-rmcat-eval-criteria]
Singh, V. and J. Ott, "Evaluating Congestion Control for Interactive Real-time Media", draft-ietf-rmcat-eval-criteria-00 (work in progress), January 2014.

[I-D.ietf-rmcat-cc-requirements]
Jesup, R., "Congestion Control Requirements For RMCAT", draft-ietf-rmcat-cc-requirements-02 (work in progress), February 2014.

[I-D.draft-sarker-rmcat-cellular-eval-test-cases]
Sarker, Z., "Evaluation Test Cases for Interactive Real-Time Media over Cellular Networks", <http://www.ietf.org/id/draft-sarker-rmcat-cellular-eval-test-cases-00.txt>.

11.2. Informative References

[I-D.ietf-rtcweb-use-cases-and-requirements]
Holmberg, C., Hakansson, S., and G. Eriksson, "Web Real-Time Communication Use-cases and Requirements", draft-ietf-rtcweb-use-cases-and-requirements-14 (work in progress), February 2014.

[RFC5033] Floyd, S. and M. Allman, "Specifying New Congestion Control Algorithms", BCP 133, RFC 5033, August 2007.

[RFC5166] Floyd, S., "Metrics for the Evaluation of Congestion Control Mechanisms", RFC 5166, March 2008.

[RFC5681] Allman, M., Paxson, V., and E. Blanton, "TCP Congestion Control", RFC 5681, September 2009.

[SA4-EVAL]
R1-081955, 3GPP., "LTE Link Level Throughput Data for SA4 Evaluation Framework", 3GPP R1-081955, 5 2008.

[SA4-LR] S4-050560, 3GPP., "Error Patterns for MBMS Streaming over UTRAN and GERAN", 3GPP S4-050560, 5 2008.

[xiph-seq]
Xiph.org, , "Video Test Media", http://media.xiph.org/video/derf/ , .

[HEVC-seq]
HEVC, , "Test Sequences", http://www.netlab.tkk.fi/˜varun/test_sequences/ , .

[TCP-eval-suite]
Lachlan, A., Marcondes, C., Floyd, S., Dunn, L., Guillier, R., Gang, W., Eggert, L., Ha, S., and I. Rhee, "Towards a Common TCP Evaluation Suite", Proc. PFLDnet. 2008, August 2008.

Appendix A. List of Network Parameters

In addition to the recommended evaluation settings in Section 4, the implemntors can extend their tests by choosing from the following values:

A.1. One-way Propagation Delay

Experiments are expected to verify that the congestion control is able to work in challenging situations, for example over trans-continental and/or satellite links. Typical values are:

1. Very low latency: 0-1ms

2. Low latency: 50ms

3. High latency: 150ms

4. Extreme latency: 300ms

A.2. End-to-end Loss

To model lossy links, the experiments can choose one of the following loss rates, the fractional loss is the ratio of packets lost and packets sent.

1. no loss: 0%

2. 1%

3. 5%

4. 10%

5. 20%

A.3. DropTail Router Queue Length

The router queue length is measured as the time taken to drain the FIFO queue. It has been noted in various discussions that the queue length in the current deployed Internet varies significantly. While the core backbone network has very short queue length, the home gateways usually have larger queue length. Those various queue lengths can be categorized in the following way:

1. QoS-aware (or short): 70ms

2. Nominal: 300-500ms

3. Buffer-bloated: 1000-2000ms

Here the size of the queue is measured in bytes or packets and to convert the queue length measured in seconds to queue length in bytes:

QueueSize (in bytes) = QueueSize (in sec) x Throughput (in bps)/8

Appendix B. Models

B.1. Jitter models

This section defines jitter model for the purposes of this document. When jitter is to be applied to both the RMCAT flow and any competing flow (such as a TCP competing flow), the competing flow will use the jitter definition below that does not allow for re-ordering of packets on the competing flow (see NR-RBPDV definition below).

Jitter is an overloaded term in communications. Its meaning is typically associated with the variation of a metric (e.g., delay) with respect to some reference metric (e.g., average delay or minimum delay). For example, RFC 3550 jitter is a smoothed estimate of jitter which is particularly meaningful if the underlying packet delay variation was caused by a Gaussian random process.

Because jitter is an overloaded term, we instead use the term Packet Delay Variation (PDV) to describe the variation of delay of individual packets in the same sense as the IETF IPPM WG has defined PDV in their documents (e.g., RFC 3393) and as the ITU-T SG16 has defined IP Packet Delay Variation (IPDV) in their documents (e.g., Y.1540).

Most PDV distributions in packet network systems are one-sided distributions (the measurement of which with a finite number of measurement samples result in one-sided histograms). In the usual packet network transport case there is typically one packet that transited the network with the minimum delay, then a majority of packets also transit the system within some variation from this minimum delay, and then a minority of the packets transits the network with delays higher than the median or average transit time (these are outliers). Although infrequent, outliers can cause significant deleterious operation in adaptive systems and should be considered in RMCAT adaptation designs.

In this section we define two different bounded PDV characteristics, 1) Random Bounded PDV and 2) Approximately Random Subject to No-Reordering Bounded PDV.

Random Bounded PDV (RBPDV):

The RBPDV probability distribution function (pdf) is specified to be of some mathematically describable function which includes some practical minimum and maximum discrete values suitable for testing. For example, the minimum value, x_min, might be specified as the minimum transit time packet and the maximum value, x_max, might be idefined to be two standard deviations higher than the mean.

Since we are typically interested in the distribution relative to the mean delay packet, we define the zero mean PVD sample, z(n), to be z(n) = x(n) - x_mean, where x(n) is a sample of the RBPDV random variable x and x_mean is the mean of x.

We assume here that s(n) is the original source time of packet n and the post-jitter induced emmission time, j(n), for packet n is j(n) = {[z(n) + x_mean] + s(n)}. It follows that the separation in the post-jitter time of packets n and n+1 is {[s(n+1)-s(n)] - [z(n)-z(n+1)]}. Since the first term is always a positive quantity, we note that packet reordering at the receiver is possible whenever the second term is greater than the first. Said another way, whenever the difference in possible zero mean PDV sample delays (i.e., [x_max-x_min]) exceeds the inter-departure time of any two sent packets, we have the possibility of packet re-ordering.

There are important use cases in real networks where packets can become re-ordered such as in load balancing topologies and during route changes. However, for the vast majority of cases there is no packet re-ordering because most of the time packets follow the same path. Due to this, if a packet becomes overly delayed, the packets after it on that flow are also delayed. This is especially true for mobile wireless links where there are per-flow queues prior to base station scheduling. Owing to this important use case, we define another PDV profile similar to the above, but one that does not allow for re-ordering within a flow.

Approximately Random Subject to No-Reordering Bounded PDV (NR-RPVD):

No Reordering RPDV, NR-RPVD, is defined similarly to the above with one important exception. Let serial(n) be defined as the serialization delay of packet n at the lowest bottleneck link rate (or other appropriate rate) in a given test. Then we produce all the post-jitter values for j(n) for n = 1, 2, ... N, where N is the length of the source sequence s to be jittered. The exception can be stated as follows: We revisit all j(n) beginning from index n=2, and if j(n) is determined to be less than [j(n-1)+serial(n-1)], we redefine j(n) to be equal to [j(n-1)+serial(n-1)] and continue for all remaining n (i.e., n = 3, 4, .. N). This models the case where the packet n is sent immediately after packet (n-1) at the bottleneck link rate. Although this is generally the theoretical minimum in that it assumes that no other packets from other flows are in-between packet n and n+1 at the bottleneck link, it is a reasonable assumption for per flow queuing.

We note that this assumption holds for some important exception cases, such as packets immediately following outliers. There are a multitude of software controlled elements common on end-to-end Internet paths (such as firewalls, ALGs and other middleboxes) which stop processing packets while servicing other functions (e.g., garbage collection). Often these devices do not drop packets, but rather queue them for later processing and cause many of the outliers. Thus NR-RPVD models this particular use case (assuming serial(n+1) is defined appropriately for the device causing the outlier) and thus is believed to be important for adaptation development for RMCAT.

[Editor’s Note: It may require to define test distributions as well. Example test distrubution may include-

1 - Two-sided: Uniform PDV Distribution. Two quantities to define: x_min and x_max.

2 - Two-sided: Truncated Gaussian PDV Distribution. Four quantities to define: the appropriate x_min and x_max for test (e.g., +/- two sigma values), the standard deviation and the mean.

3 - One Sided: TBD]

B.2. Loss generation model

[Editor’s note : Describes the model for generating packet losses, for example, losses can be generated using traces, or using the Gilbert-Elliot model, or randomly (uncorrelated loss).]

B.3. TCP taffic model

Long-lived TCP flows will download data throughout the session and are expected to have infinite amount of data to send or receive.

Each short TCP flow is modeled as a sequence of file downloads interleaved with idle periods. Not all short TCPs start at the same time, i.e., some start in the ON state while others start in the OFF state.

The short TCP flows can be modelled in two ways, 1) 100s of flows fetching small (5-20 KB) amounts of data, or 2) 10s of flows fetching slightly larger (100-1000KB) amounts of data.

The idle period is typically derived from an exponential distribution with the mean value of 10 seconds.

[Open issue: short-lived/bursty TCP cross-traffic parameters are still to be agreed upon].

Authors' Addresses

Zaheduzzaman Sarker
Ericsson AB
Luleae, SE 977 53
Sweden

Phone: +46 10 717 37 43
Email: zaheduzzaman.sarker@ericsson.com

Varun Singh
Aalto University
School of Electrical Engineering
Otakaari 5 A
Espoo, FIN 02150
Finland

Email: varun@comnet.tkk.fi
URI: http://www.netlab.tkk.fi/˜varun/

Xiaoqing Zhu
Cisco Systems
510 McCarthy Blvd
Milpitas, CA 95134
USA

Email: xiaoqzhu@cisco.com

Michael A. Ramalho
Cisco Systems, Inc.
8000 Hawkins Road
Sarasota, FL 34241
USA

Phone: +1 919 476 2038
Email: mramalho@cisco.com